



Guide de l'utilisateur

AWS Elemental MediaTailor



AWS Elemental MediaTailor: Guide de l'utilisateur

Copyright © 2026 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 c'est AWS Elemental MediaTailor ?	1
Configuration requise pour le serveur Origin	1
MediaTailor concepts	2
Types de listes de lecture HLS	2
Types de manifestes DASH	4
Concepts d'insertion de publicités	10
Concepts d'assemblage de canaux	11
Comment fonctionne l'insertion d'annonces	12
Flux d'événements d'insertion d'annonces	14
Services connexes	17
Accès MediaTailor	18
Tarification	18
Régions	18
Configuration	19
Inscrivez-vous pour Compte AWS	19
Commencer avec MediaTailor	20
Commencer à insérer MediaTailor des publicités	20
Conditions préalables	21
Étape 1 : Accès MediaTailor	21
Étape 2 : Préparer un stream	21
Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête	24
Étape 4 : créer une configuration	25
Étape 5 : Testez la configuration	27
Étape 6 : envoyez la demande de lecture à MediaTailor	28
Étape 7 (Facultatif) : Surveiller l'activité	29
Étape 8 : nettoyer	30
Débuter avec l'assemblage de MediaTailor canaux	31
Conditions préalables	32
Étape 1 : Création d'un emplacement source	32
Étape 2 : Ajoutez des sources de VOD à votre localisation	33
Étape 3 : Création d'une chaîne	34
Étape 4 : Ajoutez des programmes à la programmation de votre chaîne	36
Étape 5 (facultatif) : Utilisez MediaTailor pour insérer des publicités personnalisées dans votre flux	38

Étape 6 : Démarrez votre chaîne	39
Étape 7 : Testez votre chaîne	39
Étape 8 : nettoyer	39
Insérer des publicités	40
Codecs audio et vidéo pris en charge	41
Comprendre le comportement d'insertion des publicités	41
Comportement d'assemblage publicitaire pour la VOD	42
Comportement d'assemblage des publicités en direct	45
Vue d'ensemble de l'insertion d'annonces guidée par le serveur	48
Activer dans la configuration de lecture	48
Création d'une session guidée par le serveur	48
Suivi des publicités avec SGAI	50
Prélecture guidée avec battements de cœur manifestes	50
Compatibilité des fonctionnalités SGAI	51
Configuration en direct de SGAI	54
Configuration SGAI VOD	57
Exigences relatives à l'intégration des serveurs publicitaires	62
Des exigences VASTES	62
Exigences relatives au VPAID	65
Buffet publicitaire	66
Exemple : VAST	66
Concepts clés	67
Avantages	68
Activation	68
Comment ça marche	71
Comportement du wrapper VAST	72
Métriques et surveillance	79
Changements apportés aux événements ADS	81
Consultez aussi	82
Gestion de la configuration	83
Création d'une configuration	83
Affichage des détails de configuration	94
Modification des paramètres de configuration	94
Supprimer des configurations	94
Intégrer une source de contenu	95
Exigences relatives aux sources d'entrée	95

Intégrer une source HLS	96
Intégrer une MPEG-DASH source	108
Sécurisation des interactions d'origine avec SIGv4	131
Intégration aux plateformes publicitaires de Google	135
Server-side intégration	137
Client-side intégration	138
Optimisation du taux de remplissage des publicités	139
Personnalisation du comportement des interruptions de publicité grâce à la suppression des interruptions de publicité	142
Configuration de la suppression des interruptions publicitaires	142
Pare-chocs et insertion	156
Configuration des pare-chocs	157
Insertion d'annonces avant le lancement	158
Insertion publicitaire sur ardoise	159
Configuration de l'ardoise	160
Comment fonctionne le seuil de personnalisation	161
Configuration de Slate et VPAID	164
Transcodage des publicités	165
Transcodage dynamique	165
Profils de transcodage personnalisés	168
Publicités de prélecture	172
Comment fonctionne le préchargement	173
Création de plannings de prélecture	178
TPS-based modelage du trafic	185
Supprimer des programmes de prélecture	186
Annonces préconditionnées	187
Exigences relatives aux annonces préconditionnées	188
Flux de travail publicitaire préconditionné	191
Fonctions de monétisation	192
Comment les fonctions s'intègrent dans le MediaTailor flux de travail	193
Guide de démarrage rapide	193
Hooks de cycle de vie	198
Types de fonctions et composition	206
Création et gestion	222
Référence d'expression JSonata	225
Exemples de fonctions	236

Résolution des problèmes et surveillance	240
Restrictions	250
Variables publicitaires dynamiques	253
Variables de session	255
Variables relatives aux joueurs	269
Variables de domaine	271
Variables de service	272
Alias de configuration	281
Transmettre les paramètres ADS	290
Routage des paramètres	293
Intégration MediaPackage	295
Comportement des sessions	297
Référence des paramètres	299
Résolution des problèmes de paramètres	300
Résolution des problèmes d'alias	305
MediaTailor paramètres de requête manifeste	308
Paramètres de requête d'origine	309
MediaTailorrestrictions relatives aux caractères	310
MediaTailorSessions HLS implicites	312
MediaTailor Sessions implicites DASH	313
MediaTailor initialisation explicite de session	315
MediaTailorcomportement spécifique au protocole	317
MediaTailor Intégration au CDN	320
Rapports et données de suivi	321
Server-side suivi	322
Client-side suivi	339
Superposer des publicités	418
Conditions préalables à l'utilisation des annonces incrustées	419
Prise en main	419
Journalisation et statistiques	433
Facturation des publicités incrustées dans MediaTailor	435
Décoration d'identification publicitaire	435
État de la session	437
Insertion de manifestes et de métadonnées publicitaires	438
Interactions avec le serveur de décision publicitaire (ADS)	462
API de suivi côté client	463

Création de flux assemblés linéaires	466
Types de flux de travail d'assemblage de canaux	467
VOD-to-live	467
Live-to-live	470
VOD + Live-to-live (Mixte)	474
Choisir le bon flux de travail	478
Étapes suivantes	479
Utilisation des emplacements sources	479
Création d'un emplacement source	480
Configuration de l'authentification pour votre emplacement source	482
Utilisation de sources de VOD	491
Utilisation de sources en direct	495
Utilisation des configurations de packages	501
Mise en cache du manifeste	501
Utilisation des chaînes	501
Création d'une chaîne	502
Utilisation de groupes de sources avec les sorties de votre chaîne	505
Supprimer une chaîne	505
Ajouter un programme	506
Création d'un programme	506
Définition des cohortes d'audience et du contenu alternatif	515
Génération de manifestes spécifiques au public	518
Insérer des publicités et des pauses publicitaires	518
Configurer l'insertion de publicités	519
SCTE-35 messages pour les pauses publicitaires	520
Décalage temporel de lecture d'une chaîne	526
Utilisation d'un CDN	528
Sélection de CDN	529
Planifier l'intégration du CDN	531
Estimation des besoins en trafic	532
Configuration des stratégies d'optimisation	533
Personnalisez la planification	535
Configuration de la surveillance et de la mise à l'échelle	538
Optimisez les coûts	539
Testez votre implémentation	541
Résoudre les problèmes courants	543

Configurer l'intégration du CDN	545
Composants et exigences d'intégration du CDN	547
Conditions préalables à l'intégration du CDN	21
Étapes de configuration de l'intégration du CDN	548
En-têtes requis pour l'intégration du MediaTailor CDN	548
Configurer les comportements de routage du CDN	550
Configurer le mappage du CDN	554
Bonnes pratiques de sécurité	557
Insertion d'annonces avec CDN	559
Ce dont vous aurez besoin	559
Avant de commencer	560
Avantages de l'intégration du CDN	561
Comprendre l'architecture du CDN	562
Configurer l'insertion de publicité de base	569
SSAI avec assemblage de canaux	573
Optimisez les performances du CDN	581
Surveillez les opérations du CDN	584
Résoudre les problèmes d'insertion de publicités avec CDNs	590
Assemblage de canaux avec CDN	594
Ce dont vous aurez besoin	594
Avant de commencer	595
Avantages de l'intégration du CDN	596
Comprendre l'architecture du CDN	597
Configuration de base	599
Configurer la base URLs	600
Mettre en œuvre l'insertion de publicités	601
Configuration de l'affichage décalé dans le temps	604
Surveillez les opérations du CDN	606
Guide d'optimisation complet	606
MediaPackage Intégration au CDN	607
Comprendre le flux de travail MediaPackage et le flux de travail du CDN	607
Conditions préalables	608
Configuration des paramètres essentiels	608
Vérification de l'intégration	611
Étapes suivantes	612
Filtrage des manifestes	613

Mise en cache du CDN	619
Performances du CDN	624
Résolution des problèmes d'intégration au CDN	630
intégration d'CloudFront	634
CloudFront Configuration de base	635
Optimisation des performances	642
Multi-Region résilience	646
Surveillance et résolution des problèmes	650
Configuration d'un CDN tiers	655
Conditions préalables	656
Configurer les règles de mise en cache du CDN	657
Vérifiez la configuration de votre CDN	660
Optimisez les performances du CDN	661
Résoudre les problèmes liés aux CDN tiers	663
Optimisation des performances du CDN	664
Optimisation de la mise en cache	665
Optimisation du routage des demandes	670
Critères de performance	671
Optimisation avancée	672
Surveillance du CDN	675
Indicateurs de performance essentiels du CDN	676
Configuration des outils de surveillance	678
Liste de contrôle de mise en œuvre	680
Rubriques en relation	680
Test d'intégration au CDN	681
Conditions préalables aux tests	681
Méthodologie de test systématique	682
Liste de contrôle préalable au déploiement	687
Référence des outils de test	688
Résoudre les problèmes liés à l'intégration du CDN	689
Liste de contrôle pour un diagnostic rapide	690
Tests d'intégration	692
Erreurs 404 manifestes	696
Problèmes liés à la livraison du manifeste	697
Problèmes de livraison par segment	699
Problèmes liés à la gestion des sessions	701

Problèmes liés au calendrier des pauses publicitaires	704
Problèmes de performances et de latence	706
Comportement incohérent des appareils	713
Préparation au dépannage	714
Guides de dépannage spécifiques au flux de travail	716
Référence d'analyse du journal d'intégration du CDN	718
Interprétation du journal CDN	718
Référence des codes d'erreur	721
Outils d'analyse des journaux	723
MediaTailor ressources de configuration de journalisation	723
CloudFormation Automatisation	725
Pourquoi utiliser AWS CloudFormation	726
Préparation au déploiement	727
Déployer le modèle	730
Utiliser les ressources déployées	731
Testez et validez votre déploiement	734
Résolution des problèmes de déploiement	735
Personnalisez le modèle	737
Référence de modèles	741
CloudFrontConfiguration de production	746
Three-origin architecture	746
Priorité et configuration du comportement du cache	748
Bénéficiez d'une assistance pour l'intégration du CDN	755
Quand passer au Support	755
Recueillez des informations pour obtenir de l'aide	756
Ressources de support	757
Meilleures pratiques en matière de dossiers de support	758
Sécurité	759
Protection des données	760
Chiffrement des données	761
Gestion des identités et des accès	761
Public ciblé	762
Authentification par des identités	762
Gestion de l'accès à l'aide de politiques	764
Comment ? AWS Elemental MediaTailor fonctionne avec IAM	765
Identity-based exemples de politiques	771

Resource-based exemples de politiques	774
AWS politiques gérées	774
Utilisation des rôles liés à un service	776
Résolution des problèmes d'identité et d'accès avec	780
Validation de conformité	783
Résilience	784
Sécurité de l'infrastructure	784
Cross-service prévention adjointe confuse	785
Journalisation et surveillance	786
CloudWatch Alarmes	786
CloudTrail journaux	786
AWS Trusted Advisor	786
Surveillance et balisage	788
Affichage des journaux	789
Journaux ADS	789
Journaux des manifestes	849
Journaux de transcodage	855
Utilisation de journaux vendus	859
Écrire des journaux dans des CloudWatch journaux	863
Contrôle du volume des journaux de session d'insertion d'annonces	872
Filtrage des journaux et des événements	875
Génération de journaux de débogage	879
Surveillance à l'aide de CloudWatch métriques	884
AWS Elemental MediaTailor CloudWatch métriques	885
AWS Elemental MediaTailor CloudWatch dimensions	902
Utilisation de métriques pour diagnostiquer les manifestes périmés	903
Enregistrement des appels d'API	906
AWS Elemental MediaTailor informations dans CloudTrail	907
Compréhension AWS Elemental MediaTailor entrées de fichier journal	908
Réception d'alertes d'assemblage de canaux	910
Résolution des problèmes liés aux alertes liées aux manifestes	915
Affichage des alertes	916
Gestion des alertes	917
Balisage de ressources	917
Ressources prises en charge	918
Restrictions liées aux étiquettes	918

Gestion des tags	918
moniteur de flux de travail	919
Composants du moniteur de flux de travail	920
Services pris en charge	921
Configuration du moniteur de flux de travail	922
Utilisation du moniteur de flux de travail	942
Dual-stack points de terminaison	945
Que sont les points de terminaison à double pile ?	945
Fonctionnalités prises en charge	945
Dual-stack format du point final	946
Utilisation des points de terminaison Dual-Stack	946
Console de gestion AWS	946
AWS Command Line Interface	946
AWS Kit SDK	946
Considérations	947
Quotas	948
Quotas relatifs à l'insertion d'annonces	948
Quotas relatifs à l'assemblage des chaînes	955
Résolution des problèmes	961
Résolution des problèmes de flux d'événements	961
Flux incomplets	961
Analyse du chronométrage	964
Problèmes courants	965
Supervision des meilleures pratiques	969
Ressources supplémentaires	970
Résolution des problèmes de lecture	971
Erreurs de client	972
Erreurs de serveur	973
Exemples	976
Déduplication des balises	977
Résolution des problèmes de paramètres	977
Résolution des problèmes liés à l'omission d'annonces	978
Symptômes	978
Ignorer les raisons	978
SAUTER NEW_CREATIVE	979
Saut du délai d'expiration de l'ADS	981

Saut d'analyse VAST	982
Ignorement de l'inadéquation de durée	983
Saut de variables de session	984
Surveillance et sauts	986
Meilleures pratiques en matière de prévention	987
Matériaux de référence	988
MediaTailor ressources	992
Historique de la documentation	994
.....	mxvii

Qu'est-ce que AWS Elemental MediaTailor?

AWS Elemental MediaTailor est un service évolutif d'insertion de publicités et d'assemblage de chaînes qui s'exécute dans le AWS cloud. Vous pouvez ainsi diffuser du contenu publicitaire ciblé aux spectateurs et créer des flux linéaires tout en préservant la qualité de diffusion dans les applications vidéo OTT (over-the-top). MediaTailor l'insertion d'annonces prend en charge le streaming HTTP en direct (HLS) Apple et le streaming adaptatif dynamique MPEG sur HTTP (DASH) pour la vidéo à la demande (VOD) et les flux de travail en direct.

AWS Elemental MediaTailor l'insertion d'annonces offre des avancées importantes par rapport aux systèmes de suivi des publicités traditionnels : les publicités sont mieux monétisées, la qualité et la résolution des vidéos sont plus homogènes, et elles sont plus faciles à gérer dans les environnements multiplateformes. MediaTailor simplifie votre flux de travail publicitaire en permettant à tous les IP-connected appareils de diffuser des publicités de la même manière que les autres contenus. Le service propose également le suivi avancé des vues publicitaires, ce qui augmente encore davantage la monétisation du contenu.

AWS Elemental MediaTailor l'assemblage de canaux est un service de manifeste uniquement qui vous permet de créer des chaînes de streaming linéaires en utilisant votre contenu de vidéo à la demande (VOD) existant. MediaTailor ne touche jamais à vos segments de contenu, qui sont diffusés directement depuis votre serveur d'origine. Il MediaTailor récupère plutôt les manifestes depuis votre origine et les utilise pour créer une fenêtre de manifeste coulissante en temps réel qui fait référence aux segments de contenu sous-jacents.

MediaTailor L'assemblage des chaînes permet de monétiser facilement votre chaîne en insérant des pauses publicitaires dans votre stream sans avoir à la conditionner à l'aide de marqueurs. SCTE-35 Vous pouvez utiliser l'assemblage de chaînes avec insertion de MediaTailor publicités ou un autre service d'insertion de publicités côté serveur.

Configuration requise pour le serveur Origin

AWS Elemental MediaTailor a des exigences spécifiques en matière de communication avec le serveur d'origine :

- Ports pris en charge : accepte MediaTailor uniquement les origines utilisant les ports HTTP et HTTPS standard :
- Port 80 pour les connexions HTTP

- Port 443 pour les connexions HTTPS

MediaTailor ne prend pas en charge les ports personnalisés pour les communications avec le serveur d'origine.

- Exigences relatives au protocole - Pour des communications sécurisées, le protocole HTTPS est requis pour certains types d'origine et certains scénarios d'authentification. Pour de plus amples informations, veuillez consulter [the section called “Intégrer une source de contenu”](#).

MediaTailor concepts

Voici un aperçu des concepts utilisés dans le guide de l'AWS Elemental MediaTailor utilisateur.

Types de listes de lecture HLS

Le HTTP Live Streaming (HLS) utilise deux principaux types de listes de lecture : les listes de lecture multivariantes et les listes de lecture multimédia. Il est essentiel de comprendre la différence entre ces types de playlist pour configurer et résoudre les problèmes liés aux MediaTailor flux de travail.

Liste de lecture multivariante

Une playlist multivariante est le fichier d'index de premier niveau d'un flux de travail de streaming HLS qui répertorie tous les rendus disponibles du contenu. Il contient des références à des listes de lecture multimédia, mais ne contient aucun segment multimédia en lui-même. La liste de lecture multivariante permet aux joueurs de sélectionner le rendu le plus approprié en fonction des conditions du réseau, des capacités de l'appareil ou des préférences de l'utilisateur.

Ce type de playlist est également connu sous plusieurs autres noms dans différents contextes :

- Playlist principale (terme obsolète)
- Manifeste principal (terme courant dans le secteur)
- Playlist principale
- Playlist principale
- Fichier d'index
- Maître M3U8

Dans MediaTailor les flux de travail, la playlist multivariante est le point d'entrée pour les demandes de lecture et c'est là que commence la personnalisation des publicités.

Exemple Exemple de playlist multivariante

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-STREAM-INF:BANDWIDTH=2665726,AVERAGE-
BANDWIDTH=2526299,RESOLUTION=960x540,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=4786126,AVERAGE-
BANDWIDTH=4547599,RESOLUTION=1280x720,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=8171787,AVERAGE-
BANDWIDTH=7768157,RESOLUTION=1920x1080,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_3.m3u8
```

Liste de lecture multimédia

Une liste de lecture multimédia contient les informations relatives au segment multimédia correspondant à un rendu spécifique (niveau de qualité) du contenu. Il inclut les informations de chronométrage URLs, les segments et les autres métadonnées nécessaires à la lecture d'un rendu unique. Chaque liste de lecture multimédia représente une combinaison différente de résolution, de débit et d'autres caractéristiques.

Ce type de playlist est également connu sous plusieurs autres noms :

- Variante de playlist
- Manifeste pour enfants (terme courant dans l'industrie)
- Chunklist
- Médias M3U8
- Playlist de rendu

Dans MediaTailor les flux de travail, les playlists multimédias sont personnalisées pour inclure à la fois des segments de contenu et des segments publicitaires dans le bon ordre.

Exemple Exemple de playlist multimédia

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:123456
#EXT-X-DISCONTINUITY-SEQUENCE:0
```

```
#EXTINF:6.0,  
segment123456.ts  
#EXTINF:6.0,  
segment123457.ts  
#EXT-X-DISCONTINUITY  
#EXTINF:5.0,  
ad-segment1.ts  
#EXTINF:5.0,  
ad-segment2.ts  
#EXT-X-DISCONTINUITY  
#EXTINF:6.0,  
segment123458.ts
```

Note

Les playlists multivariantes et multimédias sont des fichiers texte qui utilisent l'extension .m3u8. La terminologie de ces fichiers varie selon les documentations et les contextes, mais les rôles fondamentaux restent les mêmes dans l'architecture de streaming HLS.

Lors de la configuration MediaTailor, vous fournissez généralement l'URL de la liste de lecture multivariante dans l'origine de votre contenu. MediaTailor gère ensuite la personnalisation de la liste de lecture multivariante et des listes de lecture multimédia pour insérer des publicités en fonction de votre configuration.

Pour plus d'informations sur les spécifications des playlists HLS, consultez la [RFC HTTP Live Streaming](#).

Types de manifestes DASH

Le streaming adaptatif dynamique via HTTP (DASH) utilise un manifeste MPD (Media Presentation Description) pour diffuser du contenu en streaming. Il est essentiel de comprendre la structure et les composants des manifestes DASH pour configurer et résoudre les problèmes des MediaTailor flux de travail.

MPD (description de la présentation aux médias)

Le MPD est le principal fichier manifeste du streaming DASH qui décrit la structure et la disponibilité du contenu multimédia. Il contient des informations sur les périodes, les ensembles d'adaptation, les représentations et les segments qui constituent le contenu diffusé en continu.

Ce type de manifeste est également connu sous plusieurs autres noms dans différents contextes :

- Manifeste DASH
- DASH MPD
- Manifeste principal (lors de la comparaison avec HLS)
- Manifeste de présentation

Dans MediaTailor les flux de travail, le MPD est le point d'entrée pour les demandes de diffusion et c'est là que commence la personnalisation des publicités.

Exemple Exemple de manifeste MPD

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011
DASH-MPD.xsd" profiles="urn:mpeg:dash:profile:isoff-live:2011"
type="dynamic" minBufferTime="PT5.000S" maxSegmentDuration="PT2.005S"
availabilityStartTime="2020-01-01T00:00:00Z" publishTime="2020-01-01T12:30:00Z"
minimumUpdatePeriod="PT2.000S" timeShiftBufferDepth="PT5M">
  <Period id="1" start="PT0.000S">
    <AdaptationSet id="1" contentType="video" segmentAlignment="true"
    bitstreamSwitching="true" frameRate="30000/1001" maxWidth="1920" maxHeight="1080"
    par="16:9">
      <Representation id="1" mimeType="video/mp4" codecs="avc1.640028" width="1920"
      height="1080" bandwidth="5000000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
      <Representation id="2" mimeType="video/mp4" codecs="avc1.4d401f" width="1280"
      height="720" bandwidth="2800000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
      <Representation id="3" mimeType="video/mp4" codecs="avc1.4d401e" width="640"
      height="360" bandwidth="1100000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```

```

<AdaptationSet id="2" contentType="audio" segmentAlignment="true" lang="eng">
  <Representation id="4" mimeType="audio/mp4" codecs="mp4a.40.2"
audioSamplingRate="48000" bandwidth="128000">
    <SegmentTemplate timescale="48000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="96000"/>
  </Representation>
</AdaptationSet>
</Period>
</MPD>

```

Period

Une période est une section temporelle d'une présentation DASH. Chaque période contient un ou plusieurs ensembles d'adaptation et représente une période médiatique. Dans les flux de travail d'insertion d'annonces, des périodes distinctes sont généralement utilisées pour faire la distinction entre le contenu et les publicités.

Ce composant est également connu sous plusieurs autres noms :

- Segment de contenu
- Section chronologie
- Segment du programme

Dans MediaTailor les flux de travail, les périodes sont utilisées pour séparer le contenu principal du contenu publicitaire, chaque annonce étant généralement représentée par sa propre période.

Exemple Exemple de période

```

<Period id="ad-period-1" start="PT30.000S" duration="PT15.000S">
  <AdaptationSet id="1" contentType="video" segmentAlignment="true"
bitstreamSwitching="true" frameRate="30000/1001" maxWidth="1920" maxHeight="1080"
par="16:9">
    <Representation id="1" mimeType="video/mp4" codecs="avc1.640028" width="1920"
height="1080" bandwidth="5000000">
      <SegmentTemplate timescale="90000" initialization="ad1/init-stream
$RepresentationID$.m4s" media="ad1/chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
    </Representation>
  </AdaptationSet>
</Period>

```

AdaptationSet

Un AdaptationSet regroupe un ensemble de versions codées interchangeables d'un ou de plusieurs composants de contenu multimédia. Par exemple, l'un AdaptationSet peut contenir plusieurs niveaux de qualité vidéo, tandis qu'un autre peut contenir plusieurs options de langue audio.

Ce composant est également connu sous le nom de :

- Groupe de composants multimédia
- Set de streaming
- Groupe de pistes

Dans les MediaTailor flux de travail, AdaptationSets ils sont préservés lors de l'insertion des publicités afin de garantir la cohérence des types de médias entre le contenu et les publicités.

Représentation

Une représentation est une version codée spécifique du contenu multimédia dans un AdaptationSet. Chaque représentation diffère généralement en termes de débit, de résolution ou d'autres paramètres de codage, ce qui permet aux clients de sélectionner la version la plus appropriée en fonction des conditions du réseau et des capacités de l'appareil.

Ce composant est également connu sous le nom de :

- Rendu (similaire à HLS)
- Niveau de qualité
- Variante de débit
- Variante Stream

Dans MediaTailor les flux de travail, les représentations dans les périodes publicitaires sont mises en correspondance aussi étroitement que possible avec les représentations dans les périodes de contenu afin de garantir une expérience de visionnage fluide.

Segment

Un segment est une unité de données multimédia qui peut être référencée individuellement par une URL. Les segments contiennent le contenu multimédia réel (vidéo, audio, etc.) et sont référencés dans le MPD. Il existe deux principaux types de segments dans DASH :

- Segment d'initialisation : contient des informations d'initialisation pour une représentation, telles que les paramètres du codec et les informations de synchronisation.

- **Segment multimédia** : contient les données multimédia réelles pour une plage de temps spécifique au sein d'une représentation.

Dans MediaTailor les flux de travail, URLs les segments sont souvent modifiés pour pointer vers le contenu ou les fichiers multimédias publicitaires appropriés.

Note

Les manifestes DASH sont des fichiers XML qui utilisent généralement l'extension. `.mpd` La terminologie de ces fichiers et de leurs composants peut varier selon la documentation et le contexte, mais la structure fondamentale de l'architecture de streaming DASH reste la même.

Lors de la configuration MediaTailor, vous fournissez l'URL du manifeste MPD dans l'origine de votre contenu. MediaTailor gère ensuite la personnalisation du manifeste, généralement en insérant des périodes supplémentaires pour les publicités en fonction de votre configuration.

Pour plus d'informations sur les spécifications du manifeste DASH, consultez les [directives de mise en œuvre du forum industriel DASH](#).

Concepts de manifeste DASH avancés

Lorsque vous travaillez avec des manifestes DASH MediaTailor, la compréhension des concepts avancés suivants peut vous aider à configurer et à résoudre les problèmes de vos flux de travail de streaming de manière plus efficace :

Calcul de l'heure de début de période

Dans les manifestes DASH, les heures de début de période sont calculées en fonction de l'`availabilityStartTime` attribut MPD et de l'`start` attribut de la période. Pour les diffusions en direct, l'heure de début est relative à l'heure de début de disponibilité, tandis que pour le contenu VOD, elle est généralement relative au début de la présentation.

Lorsqu'il MediaTailor insère des périodes publicitaires, il calcule soigneusement les heures de début afin de garantir une transition fluide entre le contenu et les publicités. Ce calcul prend en compte :

- Heure de début de la période initiale
- La durée des périodes publicitaires précédentes

- Tout décalage temporel spécifié dans la réponse du serveur de décision publicitaire

Calculs du chronométrage avant le lancement

Les publicités préenregistrées dans les manifestes DASH nécessitent un traitement spécial, car elles apparaissent avant le début du contenu principal. MediaTailor insère les annonces d'avant-lancement sous forme de périodes distinctes au début du manifeste avec des heures de début et des durées appropriées.

Pour les annonces en avant-première, MediaTailor :

- Crée une nouvelle période pour chaque annonce préalable
- Définit l'heure de début de la première période de pré-lancement à 0
- Ajuste l'heure de début de la période du contenu principal pour tenir compte de la durée totale de toutes les annonces en avant-première

Live-to-VOD transitions

Les manifestes DASH peuvent passer du format live au format VOD (vidéo à la demande) à la fin d'une diffusion en direct. Cette transition implique de changer l'attribut `MPD` de `dynamic` à `static` et d'ajuster d'autres attributs tels que `timeShiftBufferDepth` et `minimumUpdatePeriod`.

Lorsque MediaTailor les processus se manifestent pendant live-to-VOD les transitions, cela garantit que :

- Les marqueurs publicitaires sont conservés dans le manifeste VOD
- Les heures de début des périodes sont ajustées pour maintenir un calendrier approprié
- Le manifeste reste compatible avec les clients de lecture VOD

Gestion des DRM

Les informations relatives à la gestion des droits numériques (DRM) contenues dans les manifestes DASH sont généralement incluses dans les `ContentProtection` éléments contenus dans les `AdaptationSets` représentations. MediaTailor préserve ces éléments lors de l'insertion des publicités afin de garantir que la protection du contenu reste intacte.

Lors du traitement de contenu protégé par DRM, : MediaTailor

- Conserve tous les `ContentProtection` éléments du manifeste d'origine
- Garantit que le contenu publicitaire utilise des schémas DRM compatibles, le cas échéant

- Préserve tous les attributs et éléments liés aux DRM dans le manifeste

Drapeaux SCC

Les drapeaux de contrôle du contenu supplémentaire (SCC) des manifestes DASH fournissent des informations supplémentaires sur les caractéristiques du contenu et les exigences de lecture. Ces drapeaux sont généralement inclus sous forme d'attributs ou d'éléments dans la structure MPD.

Les indicateurs SCC courants des MediaTailor processus incluent :

- Informations relatives à l'évaluation du contenu
- Fonctionnalités d'accessibilité (sous-titres codés, descriptions audio)
- Avis consultatifs relatifs au contenu
- Restrictions de lecture

MediaTailor préserve ces indicateurs pendant le traitement du manifeste afin de garantir que toutes les métadonnées du contenu sont conservées dans le manifeste personnalisé.

La compréhension de ces concepts avancés vous permet de configurer MediaTailor des performances optimales et de résoudre les problèmes susceptibles de survenir dans vos flux de travail de streaming DASH.

Concepts d'insertion de publicités

Voici un aperçu des concepts liés à l'insertion de publicités.

Publicité disponible

Unité spécifique de temps de publicité comprise dans un break publicitaire qui peut être vendue aux annonceurs. Un break publicitaire peut contenir plusieurs avantages publicitaires. Lorsqu'il MediaTailor reçoit une réponse VAST d'un serveur de décision publicitaire, il remplit ces informations avec des publicités personnalisées.

Pause publicitaire

Période pendant laquelle les publicités sont diffusées pendant la programmation. Les interruptions publicitaires peuvent se produire avant le contenu (pré-diffusion), pendant le contenu (mi-diffusion) ou après le contenu (post-diffusion). MediaTailor identifie les interruptions de publicité dans les manifestes grâce à des marqueurs tels que SCTE-35.

Serveur de décision publicitaire (ADS)

Serveur qui fournit les spécifications des spots publicitaires en fonction de critères, dont les campagnes publicitaires en cours et les préférences de l'utilisateur.

Configuration

Un objet avec MediaTailor auquel vous interagissez. La configuration contient des informations de localisation concernant le serveur d'origine et le serveur de décision publicitaire (ADS). La configuration contient également des points de terminaison qui fournissent des points d'accès entrants et sortants. MediaTailor

Transcodage dynamique

Processus qui fait correspondre le format et la qualité de la publicité au contenu vidéo principal lorsque le contenu est demandé. Le transcodage dynamique réduit les exigences de stockage et garantit que la lecture passe de manière transparente du contenu publicitaire au contenu vidéo.

Manipulation manifeste

Processus de réécriture des manifestes à partir du serveur d'origine de manière à ce que les manifestes référencent les fragments de publicité et de contenu appropriés. Les publicités sont déterminées par la réponse VAST du serveur de décision publicitaire (ADS). Au fur et à mesure que la lecture progresse, MediaTailor les annonces sont insérées ou remplacées dans le flux de contenu.

VAST et VMAP

Les modèles VAST (Video Ad Serving Template) et VMAP (Video Multiple Ad Playlist) sont des réponses XML que le serveur de décision publicitaire envoie aux demandes publicitaires provenant de MediaTailor. Les réponses indiquent quelles annonces MediaTailor insèrent dans le manifeste. VMAP inclut également le calendrier des pauses publicitaires et les avantages publicitaires qu'elles contiennent. Pour plus d'informations sur la logique derrière l'insertion de publicités MediaTailor, consultez [Comprendre le comportement d'insertion des AWS Elemental MediaTailor publicités](#). Pour plus d'informations sur la façon dont MediaTailor utilise VAST, consultez [the section called “Exigences relatives à l'intégration des serveurs publicitaires”](#).

Concepts d'assemblage de canaux

Voici un aperçu des concepts liés à l'assemblage de canaux.

Canaux

Un canal assemble vos manifestes sources en un flux linéaire. Chaque canal possède une ou plusieurs sorties qui contiennent les URL de lecture auxquelles les joueurs ont accès. Les sorties de canal correspondent aux paramètres de configuration du package que vous créez pour vos sources VOD. Une chaîne contient un calendrier qui détermine à quel moment les sources de VOD seront diffusées dans le stream de la chaîne.

Configuration du package

Une configuration de packager est une représentation de votre source VOD qui contient des caractéristiques de format packagées spécifiques. Vous associez les configurations de votre package aux sorties de canal pour créer des flux de lecture pour les formats packagés de votre source VOD, tels que HTTP Live Streaming (HLS).

Horaire

Chaque chaîne est composée de programmes organisés dans la grille horaire de la chaîne. Le calendrier détermine l'heure à laquelle les programmes seront diffusés dans le flux linéaire de la chaîne.

Emplacements des sources

Un emplacement source représente le serveur d'origine sur lequel vos actifs sont stockés. Il peut s'agir d'Amazon S3, d'un serveur HTTP, d'un réseau de diffusion de contenu (CDN) ou d'une infrastructure de packaging telle que MediaPackage.

Sources de VOD

Une source VOD représente un contenu unique, tel qu'un film ou un épisode d'une émission de télévision. Vous associez des sources de VOD à des programmes pour les ajouter au flux linéaire de votre chaîne.

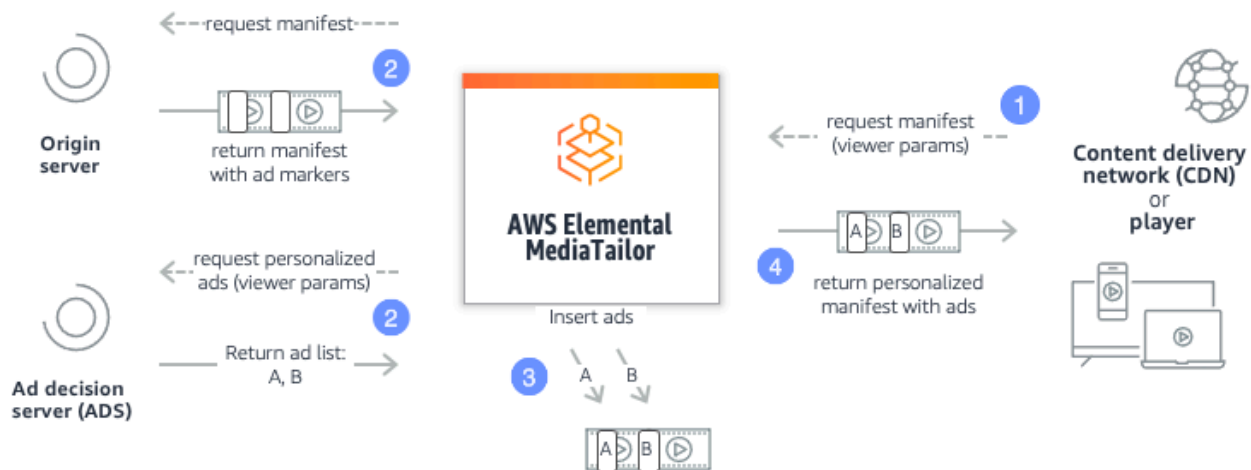
Public

Un public définit une cohorte de spectateurs qui peut éventuellement avoir un contenu alternatif. Vous pouvez définir des audiences sur des canaux linéaires standard.

Comment fonctionne MediaTailor l'insertion d'annonces

AWS Elemental MediaTailor interagit entre votre réseau de diffusion de contenu (CDN), votre serveur d'origine et votre serveur de décision publicitaire (ADS) pour intégrer des publicités personnalisées aux pauses publicitaires du contenu en direct et du contenu vidéo à la demande.

Voici un aperçu du fonctionnement de l'insertion d' annonces MediaTailor :



1. Un lecteur ou un CDN tel qu'Amazon CloudFront envoie une demande de contenu HLS ou DASH à MediaTailor . La demande contient des paramètres provenant du lecteur ainsi que des informations sur le spectateur, qui sont utilisées pour la personnalisation des publicités.
2. MediaTailor envoie une demande à l'ADS contenant les informations du spectateur. Le serveur ADS choisit les publicités en fonction des informations de l'utilisateur et des campagnes publicitaires en cours. Il renvoie les URL aux créations publicitaires dans une réponse VAST ou VMAP à MediaTailor

Si vous avez préconditionné les annonces, les URL pointent vers les annonces prétranscodées. Pour plus d'informations sur l'assemblage des annonces à l'aide d'annonces prétranscodées, consultez. [Annonces préconditionnées](#)

3. MediaTailor manipule le manifeste pour inclure les URL des annonces renvoyées par l'ADS, transcodées pour correspondre aux caractéristiques d'encodage du contenu d'origine. Si vous utilisez des annonces préconditionnées, il est de votre responsabilité de vous assurer qu'elles correspondent au modèle de manifeste.

Si une annonce n'a pas encore été transcodée pour correspondre au contenu, MediaTailor vous ne l'insérez pas et vous l'utiliserez MediaConvert pour préparer l'annonce afin qu'elle soit prête pour la prochaine demande.

4. MediaTailor renvoie le manifeste entièrement personnalisé au CDN ou au lecteur demandeur.

L'ADS suit les publicités consultées en fonction des étapes de visionnage telles que le début, le milieu et la fin de l'annonce. Au fur et à mesure de la diffusion, le lecteur MediaTailor envoie des balises de suivi publicitaire à l'URL de suivi des publicités ADS, afin d'enregistrer le nombre de vues d'une publicité. Lors de l'initialisation de la session avec MediaTailor, le joueur indique s'il doit ou MediaTailor doit envoyer ces balises pour la session.

Pour plus d'informations sur la façon de démarrer avec l'insertion d'annonces, consultez [Commencer avec MediaTailor](#).

AWS Elemental MediaTailor flux d'événements d'insertion d'annonces

AWS Elemental MediaTailor procède à la personnalisation des manifestes pour l'insertion de publicités côté serveur par le biais d'une séquence d'événements prévisible. Comprendre ce flux d'événements vous aide à comprendre comment MediaTailor transformer les opportunités publicitaires en expériences de visionnage personnalisées et fournit une base pour résoudre les problèmes en cas de problème.

Chaque opportunité d'insertion d'annonce suit une séquence chronologique d'événements enregistrés à des fins MediaTailor de visibilité et de surveillance. Ces événements constituent des étapes clés du processus de personnalisation des manifestes, qu'il s'agisse de détecter une opportunité publicitaire ou de fournir des informations de suivi.

Séquence d'événements standard

En cas de traitement MediaTailor réussi d'une opportunité d'insertion d'annonce, la séquence d'événements suivante se produit généralement :

1. Détection des opportunités publicitaires : MediaTailor détecte un marqueur publicitaire (tel que SCTE-35) dans le manifeste de contenu, indiquant qu'une opportunité de personnalisation publicitaire est disponible.
2. Demande du serveur de décision publicitaire : MediaTailor envoie une demande au serveur de décision publicitaire (ADS) configuré pour récupérer les publicités correspondant à l'opportunité détectée, y compris les paramètres du spectateur et la durée de la pause publicitaire.
3. Traitement des réponses publicitaires : MediaTailor reçoit une réponse de l'ADS contenant des informations relatives à la création publicitaire, des URL de suivi et des métadonnées relatives aux publicités à inclure. Pour plus d'informations sur les formats de réponse VAST, VMAP et VPAID, consultez. [Exigences relatives à l'intégration des serveurs publicitaires](#)
4. Personnalisation du manifeste : traite MediaTailor avec succès la réponse publicitaire, transcode les publicités si nécessaire et génère un manifeste personnalisé qui inclut les URL relatives à la

fois aux segments publicitaires et au contenu d'origine. MediaTailor envoie ensuite le manifeste personnalisé au périphérique de lecture.

5. Activation de la balise de suivi : une balise de suivi est déclenchée pour signaler les événements publicitaires (tels que les impressions, les quartiles et l'achèvement) au serveur publicitaire ou à d'autres systèmes de mesure. En mode de reporting côté serveur (par défaut), MediaTailor déclenche la balise en fonction des demandes de segments de joueurs. En mode de reporting côté client, le périphérique de lecture déclenche la balise à l'aide des URL fournies dans le manifeste personnalisé.

Durée normale : cette séquence complète prend généralement 2 à 5 secondes, en fonction du temps de réponse du serveur de décision publicitaire et des exigences en matière de traitement de création publicitaire.

Variations d'événements courantes

Toutes les tentatives d'insertion d'annonces ne suivent pas la séquence de réussite standard. MediaTailor gère différents scénarios susceptibles de modifier le flux d'événements :

Scénario de réponse publicitaire vide

Lorsque le serveur de décision publicitaire ne renvoie aucune publicité :

1. La détection des opportunités publicitaires se produit normalement
2. La demande du serveur Ad Decision est envoyée avec succès
3. ADS renvoie une réponse vide sans aucune publicité
4. Aucune publicité n'est incluse dans le manifeste personnalisé pour cette opportunité

Il s'agit d'un scénario normal qui peut se produire en raison de la disponibilité de l'inventaire publicitaire, des critères de ciblage ou des règles commerciales configurées sur le serveur de décision publicitaire. Dans ce cas, le contenu sous-jacent sera affiché, sauf s'il s'agit d'un flux en direct ou en direct à la demande et si la liste de diffusion a été configurée. Pour de plus amples informations, veuillez consulter [MediaTailor ardoise et insertion](#).

Scénarios d'erreur

Lorsque des problèmes techniques empêchent l'insertion réussie d'une annonce, le flux d'événements peut être interrompu à différents moments :

- Délai d'expiration du serveur de décision publicitaire : la demande adressée à l'ADS dépasse le délai d'expiration configuré, empêchant ainsi la récupération des publicités.
- Erreurs de communication : des problèmes de réseau ou de connectivité MediaTailor empêchent d'atteindre le serveur de décision publicitaire.
- Erreurs d'analyse des réponses - L'ADS renvoie une réponse que MediaTailor ne peut pas être analysée en raison d'un format ou d'une structure VAST non valide.

Dans les scénarios d'erreur, la lecture du contenu se poursuit MediaTailor généralement sans publicité, ce qui garantit une expérience utilisateur ininterrompue. Pour les diffusions en direct ou en direct sur vidéo à la demande avec Slate configurée, le contenu Slate peut être affiché à la place. Pour de plus amples informations, veuillez consulter [MediaTailor ardoise et insertion](#).

Scénarios de redirection VAST

Lorsque la réponse VAST initiale contient une redirection vers un autre serveur publicitaire :

1. La détection des opportunités publicitaires et la demande ADS initiale se produisent normalement
2. La réponse initiale contient une instruction de redirection
3. MediaTailor suit la redirection pour récupérer la réponse publicitaire finale
4. La personnalisation du manifeste se poursuit avec le contenu final de l'annonce
5. L'activation de la balise de suivi se produit normalement

Les redirections VAST sont courantes dans la publicité programmatique et permettent le chaînage des serveurs publicitaires et la prise de décision en temps réel. MediaTailor autorise jusqu'à 7 redirections wrapper et un nombre illimité de redirections HTTP pendant le processus de récupération des publicités.

Vue d'ensemble du calendrier des événements

Comprendre les caractéristiques temporelles des flux d'événements MediaTailor permet de définir des attentes appropriées en matière de performance d'insertion publicitaire :

- Détection d'opportunités publicitaires : survient lorsque le joueur MediaTailor reçoit une `GetManifest` demande et qu'il rencontre les marqueurs publicitaires dans le manifeste de contenu.
- Interaction avec le serveur de décision publicitaire : elle prend généralement de 100 à 500 millisecondes, selon le temps de réponse de l'ADS et les conditions du réseau.

- Personnalisation du manifeste : s'effectue généralement dans les 50 à 200 millisecondes suivant la réception de la réponse ADS.
- Synchronisation des balises de suivi : varie selon le mode de signalement. Pour les balises côté serveur, le timing est basé sur les demandes de segment du joueur client ; les balises côté client se déclenchent en fonction de l'implémentation du joueur.

Considérations relatives aux performances : le délai total entre la détection des opportunités publicitaires et la diffusion du manifeste doit généralement rester inférieur à 5 secondes afin de garantir une expérience utilisateur optimale. Les composants individuels sont terminés en quelques millisecondes, comme décrit ci-dessus. Cependant, le seuil de 5 secondes tient compte des valeurs de délai d'expiration potentielles, des nouvelles tentatives et de la variabilité du réseau qui peuvent survenir pendant le processus complet d'insertion de publicités. Des délais plus longs peuvent indiquer des problèmes de performance ADS ou de connectivité réseau.

Utilisation des connaissances relatives aux flux d'événements

La compréhension du flux d' MediaTailor événements constitue la base pour :

- Planification de la mise en œuvre - Connaître la séquence des événements permet de concevoir l'intégration des joueurs et la configuration du serveur publicitaire.
- Optimisation des performances - La compréhension des attentes en matière de délais permet d'identifier les goulots d'étranglement et les opportunités d'optimisation.
- Préparation au dépannage - La connaissance des flux d'événements normaux permet d'identifier plus facilement les problèmes.

Pour obtenir des conseils de dépannage détaillés à l'aide de l'analyse du flux d'événements, consultez [Résolution des problèmes](#). Pour obtenir des informations techniques sur la journalisation et la surveillance des événements, consultez [Affichage des journaux](#).

Services connexes

- Amazon CloudFront est un service de réseau mondial de diffusion de contenu (CDN) qui fournit des données et des vidéos en toute sécurité à vos spectateurs. Utilisez CloudFront pour diffuser du contenu avec les meilleures performances possibles. Pour plus d'informations CloudFront, consultez le [CloudFront site Web d'Amazon](#).

- AWS Elemental MediaPackage est un service d'empaquetage juste-à-temps et de montage qui personnalise les ressources vidéo en direct pour la distribution en un format compatible avec le périphérique qui effectue la demande. Utilisez AWS Elemental MediaPackage comme serveur d'origine pour préparer le contenu et ajoutez les marqueurs publicitaires avant d'envoyer les flux à MediaTailor. Pour plus d'informations sur le MediaTailor fonctionnement avec les serveurs d'origine, consultez [Comment fonctionne MediaTailor l'insertion d'annonces](#).
- Gestion des identités et des accès AWS (IAM) est un service Web qui vous permet de contrôler en toute sécurité l'accès aux AWS ressources pour vos utilisateurs. Utilisez IAM pour contrôler qui peut utiliser vos AWS ressources (authentification) et quelles ressources ils peuvent utiliser et de quelle manière (autorisation). Pour de plus amples informations, veuillez consulter [Configuration AWS Elemental MediaTailor](#).

Accès MediaTailor

Vous pouvez y accéder MediaTailor via la console du service.

Accédez à votre Compte AWS en fournissant des informations d'identification qui vérifient que vous êtes autorisé à utiliser les services.

Pour vous connecter à la MediaTailor console, utilisez le lien suivant :**<https://console.aws.amazon.com/mediatailor/home>**.

Tarification pour MediaTailor

Comme pour les autres AWS produits, il n'existe aucun contrat ni engagement minimum d'utilisation MediaTailor. Seuls les services que vous utilisez vous sont facturés. Pour en savoir plus, consultez [PricingMediaTailor](#) (Tarification).

Régions pour MediaTailor

Pour réduire la latence des données dans vos applications, MediaTailor propose des points de terminaison régionaux pour effectuer vos demandes. Pour consulter la liste des régions disponibles, consultez MediaTailor la section [Points de terminaison régionaux](#).

Configuration AWS Elemental MediaTailor

Cette section vous guide tout au long de la procédure requise pour configurer les utilisateurs afin d'accéder à AWS Elemental MediaTailor. Pour des informations générales et supplémentaires sur la gestion des identités et des accès pour MediaTailor, consultez [Gestion des identités et des accès pour AWS Elemental MediaTailor](#).

Pour commencer à l'utiliser AWS Elemental MediaTailor, procédez comme suit.

Rubriques

- [Inscrivez-vous pour Compte AWS](#)

Inscrivez-vous pour Compte AWS

Pour commencer AWS, vous avez besoin d'un Compte AWS. Pour plus d'informations sur la création d'un Compte AWS, consultez la section [Démarrage avec un Compte AWS](#) dans le Guide de Gestion de compte AWS référence.

Commencer avec MediaTailor

Pour commencer MediaTailor, vous pouvez choisir entre deux didacticiels, l'un sur la configuration de l'insertion d'annonces et l'autre sur l'assemblage de chaînes. Vous pouvez également explorer un atelier pratique pour les événements en direct. Le didacticiel d'insertion d'annonces vous guidera dans l'envoi d'une demande de lecture MediaTailor afin d'intégrer des publicités personnalisées à votre flux de contenu. Le didacticiel d'assemblage de chaînes vous montrera comment afficher le flux de votre chaîne, y compris les publicités personnalisées, directement dans un navigateur à l'aide d'une URL de lecture.

Vous pouvez également explorer le [Live Event Framework Workshop](#). Cet atelier pratique vous guide tout au long du cycle de vie de la configuration d'un événement AWS Elemental MediaTailor en direct, de la découverte initiale et du déploiement à l'optimisation et à la mise à l'échelle. Vous pouvez suivre l'atelier pour une configuration simple utilisant une MediaTailor configuration unique pour tous les événements et tous les canaux, ou pour une architecture avancée utilisant des configurations dédiées par événement ou canal.

Rubriques

- [Commencer à insérer MediaTailor des publicités](#)
- [Débuter avec l'assemblage de MediaTailor canaux](#)

Commencer à insérer MediaTailor des publicités

Pour l'utiliser AWS Elemental MediaTailor, vous devez disposer des autorisations Compte AWS et pour accéder aux MediaTailor configurations, les afficher et les modifier. Pour plus d'informations sur la procédure à utiliser, consultez [Configuration AWS Elemental MediaTailor](#).

Ce didacticiel de mise en route vous explique comment effectuer les tâches suivantes :

- Préparez vos flux de contenu HLS ou DASH
- Configuration d'un modèle d'URL de serveur de décision publicitaire (ADS)
- Création d'une MediaTailor configuration contenant un point de terminaison de lecture
- Utilisez votre lecteur ou votre réseau de diffusion de contenu (CDN) pour envoyer une demande de lecture à MediaTailor

Pour une configuration avancée des URL ADS à l'aide de variables dynamiques et de paramètres du lecteur, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour plus d'informations sur la transmission de paramètres via des CDN, consultez [MediaTailor paramètres de requête manifeste](#).

Lorsque vous aurez terminé, vous pourrez envoyer une demande de lecture MediaTailor pour que le contenu publicitaire de votre stream soit personnalisé.

Rubriques

- [Conditions préalables](#)
- [Étape 1 : Accès AWS Elemental MediaTailor](#)
- [Étape 2 : Préparer un stream](#)
- [Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête](#)
- [Étape 4 : créer une configuration](#)
- [Étape 5 : Testez la configuration](#)
- [Étape 6 : envoyez la demande de lecture à AWS Elemental MediaTailor](#)
- [Étape 7 \(facultatif\) : Surveiller AWS Elemental MediaTailor activity](#)
- [Étape 8 : nettoyer](#)

Conditions préalables

Avant de commencer, assurez-vous d'avoir terminé les étapes de [Configuration AWS Elemental MediaTailor](#).

Étape 1 : Accès AWS Elemental MediaTailor

À l'aide de vos informations d'identification IAM, connectez-vous à la MediaTailor console à l'adresse **<https://console.aws.amazon.com/mediatailor/home>**.

Étape 2 : Préparer un stream

Configurez votre serveur d'origine pour produire des manifestes pour HLS ou DASH qui sont compatibles avec AWS Elemental MediaTailor. Votre serveur d'origine doit utiliser des ports HTTP standard (port 80) ou HTTPS (port 443). MediaTailor ne prend pas en charge les ports personnalisés pour les communications avec le serveur d'origine.

Préparer un flux HLS

Les manifestes HLS doivent satisfaire aux exigences suivantes :

- Les manifestes doivent être accessibles sur l'Internet public.
- Les manifestes doivent être en direct ou VOD.
- Les manifestes doivent avoir un EXT-X-VERSION égal ou supérieur à 3.
- Pour le contenu en direct, les manifestes doivent contenir des marqueurs pour délimiter les diffusions de publicités. Cette option est facultative pour le contenu VOD, qui peut utiliser à la place les décalages VMAP.

Le fichier manifeste doit avoir les emplacements publicitaires marqués avec l'une des actions suivantes :

- #EXT-X-CUE-OUT /# EXT-X-CUE-IN (plus courant) avec des durées comme indiqué dans l'exemple suivant.

```
#EXT-X-CUE-OUT:60.00  
#EXT-X-CUE-IN
```

- #EXT-X-DATERANGE (moins courant) avec des durées comme indiqué dans l'exemple suivant.

```
#EXT-X-DATERANGE:ID="",START-DATE="",DURATION=30.000,SCTE35-OUT=0xF  
#EXT-X-DATERANGE:ID="",START-DATE="",DURATION=30.000,SCTE35-OUT=0xF
```

Tous les champs affichés pour #EXT-X-DATERANGE sont obligatoires.

La façon dont vous configurez les marqueurs de publicité dans le manifeste influence si les publicités sont insérées dans un flux ou remplacent d'autres fragments dans le flux. Pour de plus amples informations, veuillez consulter [the section called "Comprendre le comportement d'insertion des publicités"](#).

- Les listes de lecture multivariantes HLS doivent respecter la spécification HLS documentée sur [HTTP live streaming : Master playlist tags](#). En particulier, #EXT-X-STREAM-INF doit inclure les champs RESOLUTION, BANDWIDTH et CODEC.

Après avoir configuré le flux, notez le préfixe de l'URL d'origine du contenu pour la playlist multivariante. Vous en aurez besoin pour créer la configuration dans AWS Elemental MediaTailor, plus loin dans ce didacticiel.

Préparer un flux DASH

Les manifestes DASH doivent satisfaire aux exigences suivantes :

- Les manifestes doivent être accessibles sur l'Internet public.
- Les manifestes doivent être en direct ou VOD.
- Les manifestes doivent marquer les événements comme des diffusions de publicités à l'aide de marqueurs d'insertion de jointure ou marqueurs de signal temporel. Vous pouvez fournir les marqueurs de publicité en XML clair ou en binaire encodé en base64. Pour une insertion de jointure, l'indicateur de saturation de réseau doit être activé. Pour les marqueurs de signaux temporels, l'ID du type de segmentation, situé à l'intérieur de l'UPID de segmentation, doit être une valeur de coupure reconnue par AWS Elemental MediaTailor. La diffusion de publicités commence au début de l'événement et dure pendant la durée de l'évènement, si elle est spécifiée, ou jusqu'à ce que le prochain événement démarre.

L'exemple suivant illustre un événement désigné comme une diffusion de publicités à l'aide de marqueurs d'insertion de jointure. La durée de cette diffusion de publicités correspond la durée de l'événement.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></
scte35:Program>
            <scte35:BreakDuration autoReturn="true" duration="1350000"/>
          </scte35:SpliceInsert>
        </scte35:SpliceInfoSection>
      </Event>
    </EventStream>
  <AdaptationSet mimeType="video/mp4"
    ...
```



```
</AdaptationSet>  
</Period>
```

- Les diffusions de publicités doivent avoir les mêmes paramètres `AdaptationSet` et `Representation` que les flux de contenu. AWS Elemental MediaTailor utilise ces paramètres pour transcoder les publicités pour correspondre aux flux de contenu, pour une bascule en douceur entre les deux.

Une fois que vous avez configuré le flux, notez le préfixe d'URL de l'origine du contenu du manifeste DASH. Vous en aurez besoin pour créer la configuration dans AWS Elemental MediaTailor, plus loin dans ce didacticiel.

Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête

Pour déterminer les paramètres de requête dont ADS a besoin, générez une URL de balise publicitaire à partir d'ADS. Cette URL fonctionne comme modèle pour les demandes adressées à ADS et se compose des éléments suivants :

- Valeurs statiques
- Valeurs générées par AWS Elemental MediaTailor (indiquées par `session` ou par des paramètres de `avail` requête)
- Les valeurs générées par les lecteurs, obtenues à partir de l'application cliente (indiquées par les paramètres de requête `player_params`.)

Exemple URL de balise publicitaire provenant d'un serveur ADS

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=[session.id]&cust_params=[player_params.cust_params]
```

Où :

- `output` et `content_id` sont des valeurs statiques
- `PlayerSession= [session.id]` est une valeur dynamique fournie par AWS Elemental MediaTailor. La valeur de `[session.id]` change pour chaque session de lecteur et génère une autre URL pour la demande VAST de chaque session.

- `cust_params` sont des valeurs dynamiques fournies par le lecteur

La demande de playlist multivariante émanant du lecteur doit fournir des paires clé-valeur correspondant aux paramètres de `player_params`. requête de l'URL de la demande ADS. Pour plus d'informations sur la configuration de paires clé-valeur dans la demande à AWS Elemental MediaTailor, consultez. [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#)

Entrez l'URL du « modèle » configuré lorsque vous créez le server/ADS mappage d'origine dans MediaTailor, dans [Étape 4 : créer une configuration](#).

Tests

Vous pouvez utiliser une réponse VAST statique à partir de votre ADS à des fins de test. Idéalement, la réponse VAST renvoie une restitution MP4 de qualité au format mezzanine que AWS Elemental MediaTailor peut transcoder. Si la réponse de l'ADS contient plusieurs rendus de lecture, sélectionnez le rendu MP4 MediaTailor de la plus haute qualité et de la plus haute résolution et l'envoie au transcodeur.

Étape 4 : créer une configuration

La AWS Elemental MediaTailor configuration contient des informations de mappage pour le serveur d'origine et ADS.

Pour créer une configuration (console)

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Sur la page Configurations, choisissez Create configuration (Créer une configuration).
3. Dans la section Configuration en bas de la page, pour Configuration name (Nom de configuration), entrez un nom unique qui décrit la configuration. Le nom est l'identifiant principal de la configuration. La longueur maximale autorisée est de 512 caractères.
4. Pour Source de contenu vidéo, entrez le préfixe d'URL de la playlist multivariante HLS ou du manifeste DASH pour ce flux, moins l'ID de ressource. Par exemple, si l'URL de la playlist multivariante est `http://origin-server.com/a/master.m3u8`, vous devez entrer `http://origin-server.com/a/`. Vous pouvez également saisir un préfixe plus court tel que `http://origin-server.com`, mais le `/a/` doit alors être inclus dans l'ID de ressource de la demande de contenu du lecteur. La longueur maximale est de 512 caractères.

 Note

Si l'origine de votre contenu utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. (Ce ne peut pas être un certificat auto-signé.) Sinon, il AWS Elemental MediaTailor ne parvient pas à se connecter à l'origine du contenu et ne peut pas diffuser de manifestes en réponse aux demandes des joueurs.

5. Pour le serveur de décision publicitaire, entrez l'URL de votre ADS. Il s'agit de l'URL avec les variables comme décrit dans [Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête](#) ou l'URL VAST statique que vous utilisez à des fins de test. La longueur maximale est de 25 000 caractères.

 Note

Si votre ADS utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. (Ce ne peut pas être un certificat auto-signé.) La même chose s'applique également aux URL des publicités au format mezzanine renvoyées par l'ADS. Dans le cas contraire, MediaTailor ne peut pas récupérer et réassocier les publicités dans les manifestes à partir de l'origine du contenu.

6. (Facultatif si nécessaire pour DASH) Dans Localisation, choisissez DÉSACTIVÉ si des règles de routage CDN sont configurées pour accéder aux MediaTailor manifestes et si vous utilisez des rapports côté client ou si vos joueurs acceptent les redirections HTTP persistantes.

Pour en savoir plus sur la fonction Location, consultez [the section called “Fonctionnalité de localisation”](#).

7. (Facultatif) Si votre serveur d'origine produit des manifestes DASH à période unique, choisissez DASH mpd manifest origin type (Type de manifeste d'origine DASH mpd), puis SINGLE_PERIOD. Par défaut, MediaTailor gère les manifestes DASH comme des manifestes à plusieurs périodes. Pour de plus amples informations, veuillez consulter [the section called “Intégrer une MPEG-DASH source”](#).
8. Choisissez Créer une configuration.

AWS Elemental MediaTailor affiche la nouvelle configuration sur la page Configurations.

Étape 5 : Testez la configuration

Une fois que vous avez enregistré la configuration, testez le flux à l'aide d'une URL au format approprié pour votre protocole de streaming :

- Exemple : HLS

```
playback-endpoint/v1/master/hashed-account-id/origin-id/master.m3u8
```

- Exemple : DASH

```
playback-endpoint/v1/dash/hashed-account-id/origin-id/manifest.mpd
```

Où :

- `playback-endpoint` est le point de terminaison de lecture unique généré par AWS Elemental MediaTailor lorsque la configuration a été créée.

Exemple

```
https://777788889999.mediatailor.us-east-1.amazonaws.com
```

- `hashed-account-id` est votre Compte AWS identifiant.

Exemple

```
777788889999
```

- `origin-id` est le nom que vous avez attribué lors de la création de la configuration.

Exemple

```
myOrigin
```

- `master.m3u8` ou `manifest.mpd` est le nom du fichier manifeste du flux de test, ainsi que son extension de fichier. Définissez ceci afin que vous obteniez un manifeste entièrement identifié lorsque vous l'ajoutez à la source du contenu vidéo que vous avez configuré dans [the section called “Étape 4 : créer une configuration”](#).

À l'aide des valeurs des exemples précédents, les URL complètes sont les suivantes.

- Exemple : HLS

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8
```

- Exemple : DASH

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/  
AKIAIOSFODNN7EXAMPLE/myOrigin/manifest.mpd
```

Vous pouvez tester le flux à l'aide de l'une des méthodes suivantes.

- Comme illustré dans l'exemple précédent, saisissez l'URL dans un lecteur autonome.
- Testez le flux dans votre propre environnement de lecteur.

Étape 6 : envoyez la demande de lecture à AWS Elemental MediaTailor

Configurez le lecteur en aval ou le réseau de diffusion de contenu (CDN) pour envoyer les demandes de lecture au point de terminaison de lecture de la configuration, fourni à partir de AWS Elemental MediaTailor. Toutes les variables dynamiques définies par le lecteur que vous avez utilisées dans l'URL de la demande ADS [Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête](#) doivent être définies dans la demande de manifeste du lecteur.

Exemple

Supposons que l'URL ADS de votre modèle est la suivante.

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=[session.id]&cust_params=[player_params.cust_params]
```

Définissez ensuite `[player_params.cust_params]` dans la demande du joueur en faisant précéder la paire clé-valeur par `ads.` AWS Elemental MediaTailor transmet les paramètres qui ne sont pas précédés `ads.` au serveur d'origine au lieu de l'ADS.

L'URL de la demande du lecteur est une variante des exemples HLS et DASH suivants.

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8?ads.cust_params=viewerinfo
```

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/AKIAIOSFODNN7EXAMPLE/  
myOrigin/manifest.mpd?ads.cust_params=viewerinfo
```

Lors de la AWS Elemental MediaTailor réception de la demande du joueur, il définit les variables du joueur en fonction des informations contenues dans la demande. L'URL de la demande ADS résultante est une variante de ce qui suit.

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=<filled_in_session_id>&cust_params=viewerinfo
```

Pour plus d'informations sur la configuration des paires clé-valeur à transmettre au serveur ADS, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

Étape 7 (facultatif) : Surveiller AWS Elemental MediaTailor activity

Utilisez Amazon CloudWatch et Amazon CloudWatch Logs pour suivre AWS Elemental MediaTailor l'activité, notamment le nombre de demandes, d'erreurs et de publicités disponibles.

Si c'est la première fois que vous utilisez CloudWatch with AWS Elemental MediaTailor, créez un rôle Gestion des identités et des accès AWS (IAM) pour permettre la communication entre les services.

Pour autoriser AWS Elemental MediaTailor accès à CloudWatch (console)

1. Ouvrez la console IAM à l'adresse <https://console.aws.amazon.com/iam/>.
2. Dans le volet de navigation de la console IAM, sélectionnez Rôles, puis Créer un rôle.
3. Choisissez le type Autre Compte AWS rôle.
4. Pour l'ID de compte, entrez votre Compte AWS identifiant.
5. Sélectionnez Demander un ID externe et entrez **midas**. Cette option ajoute automatiquement une condition à la stratégie d'approbation qui permet au service d'assumer le rôle uniquement si la demande inclut l'élément `sts:ExternalID` approprié.
6. Choisissez Suivant : Autorisations.
7. Ajoutez une stratégie d'autorisations qui spécifie les actions que ce rôle peut effectuer. Sélectionnez l'une des options suivantes, puis choisissez Next : Review :

- CloudWatchLogsFullAccess pour fournir un accès complet à Amazon CloudWatch Logs
 - CloudWatchFullAccess pour fournir un accès complet à Amazon CloudWatch
8. Pour Nom du rôle, saisissez **MediaTailorLogger**, puis choisissez Créer un rôle.
 9. Sur la page Rôles, sélectionnez le rôle que vous venez de créer.
 10. Modifiez la relation d'approbation pour mettre à jour le mandataire :
 1. Sur la page Récapitulatif du rôle, choisissez l'onglet Relation d'approbation.
 2. Choisissez Modifier la relation d'approbation.
 3. Dans le document de stratégie, remplacez le mandataire par le service AWS Elemental MediaTailor . Il doit ressembler à ce qui suit.

```
"Principal": {  
  "Service": "mediatailor.amazonaws.com"  
},
```

L'ensemble de la stratégie doit se présenter comme suit.

4. Choisissez Mettre à jour la politique d'approbation.

Étape 8 : nettoyer

Pour éviter des frais externes, supprimez toutes les configurations superflues.

Pour supprimer une configuration (console)

1. Sur la page AWS Elemental MediaTailor Configurations, effectuez l'une des opérations suivantes :
 - Choisissez le Nom de configuration correspondant à la configuration que vous souhaitez supprimer.
 - Dans la colonne Configuration name (Nom de configuration), cochez la case d'option, puis choisissez Delete (Supprimer).
2. Dans la zone de confirmation Delete configuration (Supprimer la configuration), entrez **Delete**, puis choisissez à nouveau Delete (Supprimer).

AWS Elemental MediaTailor supprime la configuration.

Débuter avec l'assemblage de MediaTailor canaux

Ce didacticiel de mise en route vous explique comment effectuer les tâches suivantes :

- Créez un emplacement source et ajoutez-y du contenu source
- Création d'une chaîne
- Créez une liste de programmes pour diffuser le contenu de votre chaîne selon un calendrier
- Ajoutez des publicités personnalisées au flux de la chaîne à l'aide de l'insertion d' AWS Elemental MediaTailor annonces

Lorsque vous aurez terminé, vous pourrez ouvrir un navigateur, saisir l'URL de lecture de votre chaîne et consulter le flux de votre chaîne contenant des publicités personnalisées.

Ce didacticiel explique les étapes de base pour commencer à assembler des MediaTailor canaux. Pour des informations plus détaillées, consultez [Utilisation AWS Elemental MediaTailor pour créer des flux linéaires assemblés](#).

Coût estimé

- Les frais pour une chaîne active sont de 0,10\$ de l'heure. Les chaînes inactives ne vous sont pas facturées.

Rubriques

- [Conditions préalables](#)
- [Étape 1 : Création d'un emplacement source](#)
- [Étape 2 : Ajoutez des sources de VOD à votre localisation](#)
- [Étape 3 : Création d'une chaîne](#)
- [Étape 4 : Ajoutez des programmes à la programmation de votre chaîne](#)
- [Étape 5 \(facultatif\) : Utilisez MediaTailor pour insérer des publicités personnalisées dans votre flux](#)
- [Étape 6 : Démarrez votre chaîne](#)
- [Étape 7 : Testez votre chaîne](#)
- [Étape 8 : nettoyer](#)

Conditions préalables

Avant de commencer ce didacticiel, vous devez remplir les conditions suivantes :

- Assurez-vous d'avoir suivi les étapes décrites dans [Configuration AWS Elemental MediaTailor](#).
- Vous devez disposer de ressources à la fois pour le contenu source de VOD et pour la liste d'annonces. Vous devez connaître le chemin d'accès aux manifestes des actifs.

Note

Si vous utilisez un débit adaptatif automatique (ABR) ou un codage par titre, vous devez encoder vos ressources de manière à ce que toutes les variantes aient la même longueur et comportent le même nombre de pistes secondaires. Nous vous recommandons d'utiliser un modèle de codage avec une longueur de segment minimale d'une seconde.

Étape 1 : Création d'un emplacement source

Un emplacement source représente le serveur d'origine sur lequel votre contenu est stocké. Il peut s'agir d'Amazon S3, d'un serveur Web standard, d'un réseau de diffusion de contenu (CDN) ou d'une origine d'emballage, telle que AWS Elemental MediaPackage.

MediaTailor récupère les manifestes de contenu depuis votre emplacement source et les utilise pour créer une fenêtre de manifeste glissante en direct qui fait référence aux segments de contenu sous-jacents.

Pour créer un emplacement source, effectuez la procédure suivante.

Pour créer un emplacement source

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Emplacements des sources.
3. Dans la barre de navigation, choisissez Créer un emplacement source.
4. Dans Configuration de l'emplacement source, entrez un identifiant et l'emplacement de votre contenu source :
 - Nom : identifiant de votre localisation source, tel que my-origin.

- URL de base : URL de base du serveur d'origine sur lequel votre contenu est hébergé, par exemple `https://111111111111.cloudfront.net`. L'URL doit être dans un format d'URL HTTP standard, préfixé par `http ://` ou `https ://`.

5. Choisissez Créer un emplacement source.

Étape 2 : Ajoutez des sources de VOD à votre localisation

Maintenant que vous avez défini un ou plusieurs emplacements sources pour votre chaîne, vous pouvez ajouter une ou plusieurs sources VOD. Chaque source de VOD représente un contenu unique, tel qu'un seul film, un épisode d'une émission de télévision ou un clip phare.

Vous devez créer au moins une configuration de package pour votre source VOD. Chaque configuration de package contient le format du package et les paramètres de manifeste de vos sources VOD. Vous ajoutez ensuite vos configurations de package à votre chaîne pour créer des sorties.

Vous pouvez utiliser plusieurs configurations de package pour créer différents canaux de sortie. Par exemple, si votre source de VOD est packagée à la fois en HLS et en DASH, vous pouvez créer deux configurations de package pour chaque format. Vous pouvez ensuite utiliser les groupes de sources de la configuration du package pour créer des sorties à deux canaux : une pour HLS, une pour DASH.

Pour ajouter des sources VOD et créer des configurations de packages

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Emplacements des sources.
3. Dans le volet Emplacements sources, choisissez l'emplacement source que vous avez créé au cours de la [Pour créer un emplacement source](#) procédure.
4. Choisissez Ajouter une source VOD.
5. Sous Détails de la source VOD, entrez un nom pour votre source VOD, tel que `my-example-video`.
6. Sous Configurations du package > *source-group-name* entrez les informations relatives à la configuration du package :

 Note

Les configurations de package de votre source doivent toutes avoir la même durée, telle que déterminée par le manifeste de la source. De plus, toutes les sources d'une configuration de package doivent avoir le même nombre de flux enfants. Pour répondre à ces exigences, nous vous recommandons d'utiliser un modèle de codage pour vos actifs. Nous vous recommandons d'utiliser un modèle de codage avec une longueur de segment minimale d'une seconde. MediaTailor ne prend pas en charge le streaming par titre ni le streaming automatique à débit adaptatif (ABR) car ces méthodes de codage ne respectent pas ces exigences.

- **Groupe source** : entrez un nom de groupe source qui décrit la configuration de ce package, par exemple HLS-4k. Notez ce nom ; vous y ferez référence lorsque vous créerez la sortie de votre chaîne. Pour de plus amples informations, veuillez consulter [Utilisation de groupes de sources avec les sorties de votre chaîne](#).
 - **Type** : sélectionnez le format empaqueté pour cette configuration. MediaTailor prend en charge HLS et DASH.
 - **Chemin relatif** : chemin relatif entre l'URL HTTP de base de l'emplacement source et le manifeste. Par exemple, /my/path/index.m3u8.
7. Choisissez Add source (Ajouter une source).
 8. Répétez les étapes 4 à 7 de cette procédure pour ajouter la source VOD à votre liste d'annonces.

Étape 3 : Création d'une chaîne

Une chaîne regroupe vos sources dans un flux linéaire en direct. Chaque canal contient une ou plusieurs sorties qui correspondent aux configurations de package de votre source VOD.

Vous créez d'abord une chaîne, puis vous ajoutez vos sources de VOD à la programmation de la chaîne en créant des programmes.

Pour créer un canal

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Canaux.

3. Dans la barre de navigation, choisissez **Créer une chaîne**.
4. Dans **Détails de la chaîne**, entrez les informations relatives à votre chaîne :
 - **Nom** : Entrez un nom pour votre chaîne.
 - **Mode de lecture** : détermine quels types de transitions de programme sont autorisés et ce qu'il advient d'un programme une fois qu'il est terminé. Utilisez le mode boucle par défaut.
5. Choisissez **Suivant**.
6. Dans **Détails de la sortie**, définissez les paramètres de cette sortie :
 - **Nom du manifeste** : entrez un nom de manifeste, tel que ***index***. MediaTailor ajoutera l'extension de format, telle que .m3u8 pour HLS.

 Note

Vous devez saisir un nom de manifeste unique par canal de sortie.

- **Type de format** : sélectionnez le format de diffusion pour la chaîne. DASH et HLS sont pris en charge. Choisissez le format qui correspond à la configuration du package dans laquelle vous avez créé [Étape 1 : Création d'un emplacement source](#).
 - **Groupe source** : entrez le nom du groupe source dans lequel vous avez créé [Étape 1 : Création d'un emplacement source](#).
7. Dans **Paramètres du manifeste**, entrez des informations supplémentaires concernant les paramètres de votre manifeste :
 - **Fenêtre du manifeste (sec)** : fenêtre temporelle (en secondes) contenue dans chaque manifeste. La valeur minimale est de 30 secondes et la valeur maximale est de 3 600 secondes.
 8. Choisissez **Suivant**.
 9. Sous **Politique de chaîne**, sélectionnez **Ne pas joindre la politique de chaîne**. Cette option limite la lecture aux seules personnes ayant accès à vos Compte AWS informations d'identification.
 10. Choisissez **Suivant**.
 11. Vérifiez vos paramètres dans le volet **Réviser et créer**.
 12. Choisissez **Create channel (Créer un canal)**.

 Note

Les chaînes sont créées à l'état arrêté. Votre chaîne ne sera pas active tant que vous ne l'aurez pas démarrée.

Étape 4 : Ajoutez des programmes à la programmation de votre chaîne

Maintenant que vous avez une chaîne, vous allez ajouter des programmes à la programmation de la chaîne. Chaque programme contient une source de VOD provenant d'un emplacement source de votre compte. La programmation des chaînes détermine l'ordre dans lequel vos programmes seront diffusés dans le flux de la chaîne.

Chaque programme peut comporter un ou plusieurs pauses publicitaires. Vous insérez une pause publicitaire en spécifiant une source de VOD à utiliser comme liste d'annonces. La durée de la pause publicitaire est déterminée par la durée de l'ardoise. Vous pouvez éventuellement utiliser un serveur d'insertion d'annonces côté serveur, tel que l'insertion d' annonces MediaTailor, pour personnaliser vos pauses publicitaires.

Pour ajouter des programmes à la programmation de votre chaîne

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux** > **Canaux**.
3. Dans le volet **Canaux**, choisissez le canal que vous avez créé au cours de la [Étape 3 : Création d'une chaîne](#) procédure.
4. Dans **Détails** du programme, entrez les informations relatives à votre programme :
 - **Nom** : il s'agit du nom du programme à ajouter à la programmation de votre chaîne.
 - **Nom de l'emplacement source** : choisissez **Sélectionner un emplacement source existant**, puis sélectionnez l'emplacement source que vous avez créé dans le [Étape 1 : Création d'un emplacement source](#) menu déroulant **Sélectionner un emplacement source**.
 - **Nom de la source VOD** : Choisissez **Sélectionnez une source VOD existante** et sélectionnez la source VOD que vous avez créée plus tôt dans ce didacticiel.
5. Dans **Configuration de lecture**, définissez comment et quand un programme est inséré dans la programmation d'une chaîne :

- **Type de transition** : cette valeur est fixée à **Relative**. Le type de transition relatif indique que ce programme se produit par rapport aux autres programmes de la liste des programmes.
- **Position relative** : s'il s'agit du premier programme de la programmation de votre chaîne, vous pouvez ignorer ce réglage. S'il ne s'agit pas du premier programme de la programmation de votre chaîne, choisissez où vous souhaitez l'ajouter dans la liste des programmes. Vous pouvez sélectionner **Avant le programme** ou **Après le programme**.
- **Programme relatif** : s'il s'agit du premier programme de votre emploi du temps, vous pouvez ignorer ce réglage. S'il ne s'agit pas du premier programme de la programmation de votre chaîne, choisissez **Utiliser le programme existant**, puis sélectionnez le nom du programme que vous avez créé dans [Pour ajouter des programmes à la programmation de votre chaîne](#).

6.

Sélectionnez **Ajouter une pause publicitaire**. Sous **Prises publicitaires**, configurez les paramètres de la pause publicitaire :

- **Nom de l'emplacement source de votre ardoise** : Choisissez **Sélectionnez un emplacement source existant** et choisissez l'emplacement source où est stockée votre ardoise que vous avez créé plus tôt dans ce didacticiel.
- **Nom de la source VOD** : Choisissez **Sélectionnez une source VOD existante** et choisissez la source VOD que vous utilisez pour l'ardoise que vous avez ajoutée plus tôt dans ce didacticiel. La durée de la liste détermine la durée de la pause publicitaire.
- **Pour le décalage en millisecondes** : cette valeur détermine l'heure de début de la pause publicitaire en millisecondes, en tant que décalage par rapport au début du programme. Entrez une valeur inférieure à la durée de la source VOD et qui correspond à une limite de segment pour toutes les pistes de la source VOD du programme (toutes les pistes audio, vidéo et sous-titrées), sinon la pause publicitaire sera ignorée. Par exemple, si vous saisissez **0**, cela crée une pause publicitaire avant le lancement qui sera diffusée avant le début du programme.
Remarque :.

7. Choisissez **Ajouter un programme**.

Pour plus d'informations sur les programmes, consultez [Configuring ad breaks for your program](#).

Pour obtenir des informations plus détaillées sur l'utilisation des publicités dans votre flux linéaire, consultez [Paramètres de configuration facultatifs](#).

Étape 5 (facultatif) : Utilisez MediaTailor pour insérer des publicités personnalisées dans votre flux

Vous avez maintenant une chaîne avec des programmes. Si vous le souhaitez, vous pouvez l'utiliser MediaTailor pour insérer des publicités personnalisées dans les pauses publicitaires de vos programmes dans le flux de la chaîne.

Conditions préalables

Avant de poursuivre, vous devez remplir les conditions suivantes :

- Vous devez disposer d'un serveur de décision publicitaire (ADS).
- Vous devez avoir configuré les paramètres Ad break dans la [Ajouter un programme à la programmation d'une chaîne](#) procédure.

Pour ajouter des publicités personnalisées au flux de votre chaîne à l'aide de MediaTailor

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Configurations.
3. Dans Paramètres requis, entrez les informations de base requises concernant votre configuration :
 - Nom : nom de votre configuration.
 - Source du contenu : entrez l'URL de lecture de la sortie de votre chaîne, sans le nom et l'extension du fichier. Pour des informations avancées sur MediaTailor la configuration, consultez [Réglages requis](#).
 - Ajouter un serveur de décision : entrez l'URL de votre ADS.
4. Vous pouvez éventuellement configurer les alias de configuration, les détails de personnalisation et les paramètres avancés. Pour plus d'informations sur ces paramètres, consultez [Paramètres de configuration facultatifs](#).
5. Dans la barre de navigation, choisissez Créer une configuration.

Pour des informations plus détaillées sur l'utilisation de l'insertion d' MediaTailor annonces, consultez [Utilisation AWS Elemental MediaTailor pour insérer des publicités](#).

Étape 6 : Démarrez votre chaîne

Vous avez maintenant une chaîne. Mais avant de pouvoir accéder au flux de la chaîne, vous devez démarrer votre chaîne. Si vous essayez d'accéder à une chaîne avant qu'elle ne soit active, MediaTailor renvoie un code 4xx d'erreur HTTP.

Démarrez votre chaîne

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux** > **Canaux**.
3. Dans la barre de navigation, choisissez **Démarrer**.

Étape 7 : Testez votre chaîne

Pour vérifier que votre chaîne fonctionne correctement, ouvrez un navigateur Web et saisissez l'URL de la sortie de votre chaîne. Vous devriez voir le stream de votre chaîne.

Dans certains cas, il se peut que vous ayez besoin de vider le cache pour observer le comportement attendu.

Étape 8 : nettoyer

Une fois que vous avez terminé avec la chaîne que vous avez créée pour ce didacticiel, vous devez la nettoyer en la supprimant.

Vous cesserez de payer des frais pour cette chaîne dès que le statut de la chaîne passera à Arrêtée. Pour conserver votre chaîne pour plus tard, sans encourir de frais, vous pouvez arrêter la chaîne maintenant, puis la redémarrer plus tard.

Pour supprimer votre chaîne

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux** > **Canaux**.
3. Sélectionnez la chaîne que vous souhaitez supprimer.
4. Si votre chaîne est en cours d'exécution, dans le menu déroulant **Actions**, choisissez **Arrêter**. Vous devez arrêter votre chaîne avant de pouvoir la supprimer.
5. Lorsque votre chaîne est arrêtée, dans le menu déroulant **Actions**, choisissez **Supprimer**.

Utilisation AWS Elemental MediaTailor pour insérer des publicités

Une configuration est un objet avec lequel vous interagissez dans AWS Elemental MediaTailor. La configuration contient les informations de mappage pour le serveur d'origine et le serveur de décision publicitaire (ADS). Vous pouvez également définir une lecture par défaut MediaTailor à utiliser lorsqu'une annonce n'est pas disponible ou qu'elle n'est pas disponible dans son intégralité.

Si vous utilisez un réseau de diffusion de contenu (CDN) avec MediaTailor, vous devez configurer les règles de comportement dans le CDN avant d'ajouter des informations CDN à la configuration. Pour plus d'informations sur la configuration de votre CDN, consultez [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#).

Rubriques

- [Codecs audio et vidéo pris en charge](#)
- [Comprendre le comportement d'insertion des AWS Elemental MediaTailor publicités](#)
- [MediaTailor présentation et mise en œuvre de l'insertion d'annonces guidées par le serveur](#)
- [MediaTailor exigences relatives à l'intégration des serveurs publicitaires](#)
- [Buffet publicitaire](#)
- [MediaTailor gestion de la configuration de lecture](#)
- [Intégrer une source de contenu pour l'insertion d' MediaTailor annonces](#)
- [Intégration AWS Elemental MediaTailor avec les plateformes publicitaires de Google](#)
- [Personnalisation du comportement des interruptions de publicité grâce à la suppression des interruptions de publicité](#)
- [MediaTailor pare-chocs et insertion](#)
- [MediaTailor pré-enroulement et insertion](#)
- [MediaTailor ardoise et insertion](#)
- [Transcodage des annonces dans AWS Elemental MediaTailor](#)
- [Publicités de prélecture](#)
- [Utilisation de publicités préconditionnées avec AWS Elemental MediaTailor](#)
- [Utilisation des fonctions de MediaTailor monétisation](#)
- [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#)

- [MediaTailor paramètres de requête manifeste](#)
- [Rapports et données de suivi](#)
- [Superposer des publicités](#)
- [Décoration d'identification publicitaire](#)

Codecs audio et vidéo pris en charge

MediaTailor prend en charge les codecs suivants.

- Codecs audio : mp4a, et ac-3 ec-3
- Codecs vidéo : h.264 (AVC), h.265 (), (HEVC) av01 AV1

Comprendre le comportement d'insertion des AWS Elemental MediaTailor publicités

AWS Elemental MediaTailor intègre les publicités à du contenu en direct ou à du contenu vidéo à la demande (VOD) en remplaçant ou en insérant des publicités dans le manifeste d'origine. L'insertion ou le remplacement des publicités dépend de la manière dont les pauses publicitaires sont configurées dans le manifeste d'origine et du fait que le contenu est en VOD ou en direct. Une pause publicitaire est la période pendant laquelle des publicités sont diffusées pendant la programmation, tandis que les disponibilités publicitaires sont les unités de temps publicitaires spécifiques d'une pause publicitaire qui peuvent être remplies de publicités.

- Avec le remplacement des publicités, les MediaTailor segments de contenu sont remplacés par des publicités.
- Avec l'insertion d'annonces, MediaTailor insère du contenu publicitaire là où les segments n'existent pas.

Pour plus d'informations sur la manière MediaTailor d'intégrer les publicités au contenu en direct et à la VOD, sélectionnez le sujet approprié.

Rubriques

- [Comportement d'assemblage publicitaire pour la VOD](#)
- [Comportement d'assemblage des publicités en direct](#)

Comportement d'assemblage publicitaire pour la VOD

MediaTailor insère ou remplace les publicités dans le contenu VOD en fonction de la façon dont les marqueurs publicitaires sont configurés dans le manifeste d'origine et si le serveur de décision publicitaire (ADS) envoie des réponses VMAP.

Pour un comportement des publicités par configuration des marqueurs, consultez les sections suivantes.

Si des marqueurs publicitaires sont présents

AWS Elemental MediaTailor insère des publicités où des marqueurs publicitaires SCTE-35 sont présents dans le manifeste d'origine. Les marqueurs publicitaires dont EXT-X-CUE-OUT la valeur est la 0 durée indiquent l'insertion d'une annonce.

Directives relatives aux marqueurs publicitaires HLS

Suivez ces directives pour la signalisation SCTE après le lancement et le module publicitaire :

Publicités pré-diffusées

Pour les post-déploiements, les marqueurs CUE-OUT/IN doivent précéder le dernier segment de contenu. En effet, la spécification HLS nécessite que les décorateurs de balise soient explicitement déclarés avant un segment.

Par exemple, supposons la déclaration suivante.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Videocontent.ts
#EXT-X-ENDLIST
```

AWS Elemental MediaTailor insère un post-roll comme suit.

```
#EXTINF:4.000,
Videocontent.ts
#EXT-X-DISCONTINUITY
#EXTINF:3.0,
Adsegment1.ts
#EXTINF:3.0,
```

```
Adsegment2.ts
#EXTINF:1.0,
Adsegment3.ts
#EXT-X-ENDLIST
```

Exemple 2 : Ajouter des modules

CUE-OUT/IN Les balises doivent être explicitement attachées à un segment. Vous ne pouvez pas utiliser plusieurs balises CUE-OUT/IN à la suite pour imiter un comportement de pod publicitaire.

Par exemple, la déclaration suivante est une utilisation valide de CUE-OUT/IN pour présenter un module publicitaire.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Somecontent1.ts
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Somecontent2.ts
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Videocontent.ts
```

La déclaration précédente génère une sortie similaire à la suivante.

```
Ad 1
Somecontent.ts
Ad 2
Somecontent2.ts
Videocontent.ts
Post-Roll Ad 3
```

La déclaration suivante n'est pas valide.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXT-X-CUE-OUT: 0
```

```
#EXT-X-CUE-IN  
#EXTINF:4.000,  
Videocontent.ts
```

Si aucun marqueur publicitaire n'est présent

Les marqueurs publicitaires sont le moyen recommandé pour signaler les interruptions de publicité dans un manifeste. Cependant, les marqueurs publicitaires ne sont pas obligatoires. Si le manifeste ne contient pas de marqueurs publicitaires pour DASH ou HLS, il passe MediaTailor un seul appel à l'ADS et crée des pauses publicitaires en fonction de la réponse :

- Si l'ADS envoie une réponse VAST, il MediaTailor insère toutes les annonces issues de la réponse dans un break publicitaire au début du manifeste. Il s'agit d'un pré-déploiement.
- Si l'ADS envoie une réponse VMAP, il MediaTailor utilise les décalages de temps de pause publicitaire pour créer des pauses et les insérer dans le manifeste aux heures spécifiées (avant, à mi-diffusion ou après la diffusion). MediaTailor utilise toutes les annonces de chaque pause publicitaire dans la réponse VMAP pour chaque pause publicitaire du manifeste.

Note

Lorsqu'un segment chevauche un point d'insertion avec VMAP pour du contenu VOD, MediaTailor arrondit au point d'insertion le plus proche.

Tip

Si vous souhaitez créer des coupures publicitaires en milieu de déploiement (mid-roll), mais que votre ADS ne prend pas en charge VMAP, vous devez vous assurer que des marqueurs de publicité sont présents dans le manifeste. MediaTailor insère les publicités au niveau des marqueurs, comme décrit dans les sections suivantes.

Note

Pour les méthodes d'insertion de publicités guidées par le serveur, MediaTailor insère les publicités prédiffusées en haut du manifeste et le joueur les diffuse avant les autres types d'annonces.

Comportement d'assemblage des publicités en direct

Dans les diffusions en direct, effectue AWS Elemental MediaTailor toujours le remplacement des publicités, en préservant le plus possible le temps total entre les marqueurs publicitaires. Lorsque les marqueurs publicitaires incluent l'`DURATION`attribut, MediaTailor utilise la valeur pour déterminer la durée de la pause publicitaire. Chaque CUE-OUT indicateur doit avoir une durée ou un CUE-IN indicateur correspondant dans les flux de travail en direct.

MediaTailor remplace les publicités pour le contenu en direct HLS et DASH. Pour plus d'informations sur le mode de MediaTailor calcul de l'emplacement et de la durée des pauses publicitaires, reportez-vous aux sections [the section called “Marqueurs de publicité”](#) et [the section called “Marqueurs de publicité”](#).

Sélection et remplacement des annonces

AWS Elemental MediaTailor inclut les publicités provenant de la réponse VAST du serveur de décision publicitaire (ADS) comme suit :

- Si une durée est spécifiée, MediaTailor sélectionne un ensemble d'annonces correspondant à cette durée et les inclut.
- Si aucune durée n'est spécifiée, MediaTailor diffuse autant de publicités que possible jusqu'à ce qu'un marqueur publicitaire indique un retour au contenu principal.

AWS Elemental MediaTailor respecte les directives suivantes lors du remplacement des publicités en direct :

- MediaTailor essaie de diffuser des publicités complètes, sans découpage ni troncature.
- Chaque MediaTailor fois qu'un marqueur publicitaire indique la fin de la pause publicitaire, il revient au contenu sous-jacent. Cela peut signifier la troncature d'une publicité en cours de lecture.
- À la fin de la durée, MediaTailor revient au contenu sous-jacent.
- S'il n'y a plus de publicités à diffuser pendant la durée d'une pause publicitaire, le système diffuse la liste, si une telle option est configurée, ou reprend la lecture du flux de contenu sous-jacent. Cela se produit généralement lorsqu'il n'y a pas assez de publicités transcodées pour remplir la durée de la coupure publicitaire.

 Tip

Vous pouvez définir la limite de temps publicitaire non rempli autorisé dans une coupure avec le paramètre de configuration de seuil de personnalisation. Pour plus d'informations, consultez la [PlaybackConfiguration référence](#).

Prélancement en direct pour l'insertion de publicités guidée par le serveur

Le prélancement en direct fonctionne différemment pour les méthodes d'insertion d'annonces guidées par le serveur par rapport à l'insertion d'annonces côté serveur :

Insertion de publicités côté serveur (mode cousu)

Les publicités Preroll remplacent une partie du contenu en direct au début de la session de chaque spectateur. Chaque spectateur voit le pré-lancement à des moments différents en fonction du moment où il rejoint le stream.

Méthodes d'insertion de publicités guidées par le serveur

MediaTailor place une balise de plage de dates preroll en haut de tous les manifestes multimédias contenant des attributs. CUE="PRE , ONCE" Cela oblige les joueurs à demander et à diffuser des annonces d'avant-lancement une seule fois au début de la diffusion, même s'ils partagent le même manifeste non personnalisé.

Exigences de configuration :

- Serveur de décision publicitaire Live Preroll : configurez un point de terminaison VAST pour les publicités en avant-première (cela peut être différent des publicités en cours de diffusion)
- Durée maximale autorisée de l'avant-lancement en direct : définissez la durée maximale des annonces préliminaires (facultatif, si ce paramètre est omis, toutes les publicités renvoyées seront utilisées)

Mise en œuvre technique :

- Utilisations des balises de plage de dates Preroll (époque UnixSTART-DATE="1970-01-01T00:00:00.000Z")

- Les demandes de liste d'actifs pour le pré-lancement utilisent le serveur de décision publicitaire pré-lancement configuré au lieu de l'ADS normal
- Les joueurs identifient les demandes de pré-lancement grâce `availId="aws-mediatailor-preroll-1"` aux données de la liste des actifs

Important

Pour les diffusions en direct, les annonces d'avant-lancement couvrent le contenu au lieu de le retarder. Les versions futures peuvent prendre en charge le mode de temporisation du contenu grâce à des options de configuration supplémentaires.

Note

Le comportement des pré-lancements varie entre le contenu en direct et le contenu VOD pour l'insertion de publicités guidée par le serveur. Le contenu en direct nécessite une configuration de pré-lancement explicite, tandis que le contenu VOD inclut le pré-lancement par défaut en utilisant le serveur de décision publicitaire standard.

Exemples

- Si la durée de la pause publicitaire est fixée à 70 secondes et que la réponse ADS contient deux publicités de 40 secondes, AWS Elemental MediaTailor diffuse l'une des publicités de 40 secondes. Dans le temps qui reste, il passe à la liste configurée ou au contenu sous-jacent. Au cours de ce processus, si MediaTailor rencontre un indicateur entrant, il passe aussitôt au contenu sous-jacent.
- Si la durée de la pause publicitaire est fixée à 30 secondes et que la publicité la plus courte fournie par la réponse ADS est de 40 secondes, aucune MediaTailor publicité n'est diffusée. Si une liste de publicités est configurée, elle MediaTailor est diffusée pendant 30 secondes ou jusqu'à ce qu'un indicateur apparaisse. Dans le cas contraire MediaTailor , le contenu sous-jacent est diffusé.

MediaTailor présentation et mise en œuvre de l'insertion d'annonces guidées par le serveur

AWS Elemental MediaTailor L'insertion d'annonces guidée par le serveur (SGAI) constitue une alternative à l'insertion d'annonces côté serveur en référençant les publicités sous forme de playlists distinctes plutôt que de les assembler directement dans des playlists multimédias. Cette approche améliore les performances grâce à des manifestes pouvant être mis en cache et permet une meilleure évolutivité.

Pour plus d'informations sur l'utilisation de l'insertion publicitaire guidée par le serveur avec MediaTailor, choisissez la rubrique applicable dans la liste suivante.

Activer dans la configuration de lecture

Pour permettre aux joueurs d'utiliser l'insertion publicitaire guidée par le serveur, vous devez définir Insertion Mode cette option `PLAYER_SELECT` dans la configuration de MediaTailor lecture. Cela permet aux joueurs de sélectionner l'insertion de publicités cousues ou guidées au moment de l'initialisation de la session.

Création d'une session guidée par le serveur

Lorsque vous créez des sessions de lecture, choisissez le mode guidé. La manière de procéder dépend du fait que vos joueurs utilisent des sessions implicites ou explicites.

Sessions guidées par le serveur créées implicitement

Ajouter `aws.insertionMode=GUIDED` à la demande de playlist multivariante HLS. Exemple :

```
playback-endpoint/v1/master/hashed-account-id/origin-id/index.m3u8?  
aws.insertionMode=GUIDED
```

Où :

- `playback-endpoint` est le point de terminaison de lecture unique généré par AWS Elemental MediaTailor lorsque la configuration a été créée.

Exemple

```
https://777788889999.mediatailor.us-east-1.amazonaws.com
```

- `hashed-account-id` est votre Compte AWS identifiant.

Exemple

```
777788889999
```

- `origin-id` est le nom que vous avez attribué lors de la création de la configuration.

Exemple

```
myOrigin
```

- `index.m3u8` est le nom du manifeste issu du flux de test plus son extension de fichier. Définissez ceci afin que vous obteniez un manifeste entièrement identifié lorsque vous l'ajoutez à la source du contenu vidéo que vous avez configuré dans [the section called "Étape 4 : créer une configuration"](#).

En utilisant les valeurs des exemples précédents, les valeurs complètes URLs sont les suivantes.

- Exemple :

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/777788889999/  
myOrigin/index.m3u8?aws.insertionMode=GUIDED
```

Sessions guidées par le serveur créées de manière explicite

Ajoutez `insertionMode=GUIDED` aux métadonnées JSON que le joueur envoie le HTTP au point de terminaison POST du préfixe d'initialisation de session de la MediaTailor configuration.

L'exemple suivant montre la structure des métadonnées JSON :

```
{  
  # other keys, e.g. "adsParams"  
  "insertionMode": "GUIDED"      # this can be either GUIDED or STITCHED  
}
```

Avec ces métadonnées d'initialisation, la session de lecture utilisera l'insertion de publicités guidée par le serveur.

Suivi des publicités avec SGAI

SGAI prend en charge le suivi des publicités côté serveur et côté client. Vous définissez le mode de rapport lors de l'initialisation de la session. Le mode ne peut pas changer pendant la session.

Suivi côté serveur (par défaut)

MediaTailor déclenche automatiquement des balises VAST lorsque le joueur demande des segments publicitaires. Les annonces figurant URIs dans la liste des actifs contiennent des métadonnées de balises cryptées (`awsBeaconDataawsBeaconDomain,,awsConfigurationName`). Le joueur doit accepter la substitution de `#EXT-X-DEFINE:QUERYPARAM` variables HLS. La réponse relative à la liste des actifs ne contient aucune TRACKING section.

Pour plus de détails sur le fonctionnement du balisage SGAI côté serveur, consultez [Server-side suivi avec insertion d'annonces guidée par serveur \(SGAI\)](#)

Suivi côté client

Ajoutez `aws.reportingMode=CLIENT` à votre demande d'initialisation de session. La réponse à la liste des actifs inclut une TRACKING section avec une balise URLs que le joueur déclenche pendant la diffusion de la publicité. Le point de terminaison de `GetTracking` l'API n'est pas utilisé pour les sessions SGAI. Au lieu de cela, chaque réponse à la liste des actifs inclut directement les données de suivi. Les données de suivi utilisent le même schéma JSON que la réponse de suivi SSAI (Server-side ad insertion).

Pour en savoir plus, consultez [Server-guided insertion d'annonces](#).

Prélecture guidée avec battements de cœur manifestes

Pour les diffusions SGAI en direct, vous pouvez activer la prélecture des publicités basée sur le manifeste en ajoutant `aws.guidedPrefetchMode=MANIFEST` à votre demande d'initialisation de session :

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/777788889999/myOrigin/index.m3u8?aws.insertionMode=GUIDED&aws.guidedPrefetchMode=MANIFEST
```

Lorsque cette option MediaTailor est activée, ajoute un identifiant de session (`aws.sessionId=<id>`) en tant que paramètre de requête à chaque URL du manifeste multimédia

interstitiel (/v1/i-media) de la liste de lecture multivariante. Chaque fois que le lecteur actualise un manifeste i-media, la demande reçoit l'identifiant de session, qui est MediaTailor utilisé pour mettre en file d'attente les demandes de prélecture pour les prochaines pauses publicitaires. MediaTailor

- Le `aws.guidedPrefetchMode` paramètre accepte deux valeurs : MANIFEST (activé) et OFF (désactivé, par défaut).
- Le mode de prélecture guidé n'est valide que pour les sessions SGAI. Son utilisation avec des sessions cousues renvoie une erreur.
- DASH ne prend pas encore en charge le mode de prélecture guidée.
- La prélecture guidée est indépendante du mode de rapport : les balises se déclenchent au moment de la lecture, et non au moment de la prélecture.
- Ne mettez pas en cache les manifestes i-media dans votre CDN lorsque vous utilisez le préchargement guidé. Le mécanisme de prélecture dépend de la réception directe des demandes d'actualisation du manifeste du joueur. MediaTailor Si votre CDN met en cache les /v1/i-media réponses, MediaTailor ne reçoit pas les demandes de pulsation et ne peut pas déclencher le préchargement.

MediaTailor matrice de compatibilité des fonctionnalités d'insertion publicitaire guidée par le serveur

AWS Elemental MediaTailor propose deux méthodes d'insertion d'annonces avec différentes fonctionnalités compatibles. L'insertion d'annonces guidée par le serveur fonctionne différemment de l'insertion d'annonces côté serveur, ce qui affecte la compatibilité avec certaines fonctionnalités. MediaTailor Utilisez ce tableau pour comprendre quelles fonctionnalités fonctionnent avec chaque méthode d'insertion d'annonces.

Compatibilité des fonctionnalités par méthode d'insertion d'annonces

Fonctionnalité	Insertion de publicités côté serveur (SSAI)	Insertion de publicités guidée par le serveur (SGAI)
Prélecture des annonces	✓ Soutenu	Pas encore pris en charge
Suppression des publicités	✓ Soutenu	Non applicable

Fonctionnalité	Insertion de publicités côté serveur (SSAI)	Insertion de publicités guidée par le serveur (SGAI)
Comportement des publicités avant la diffusion	Contrôlé par MediaTailor la configuration	Contrôlé par MediaTailor la configuration
Suivi des publicités côté client	Utilise GetTracking l'API	TRACKINGSection Utilise dans la liste des actifs (GetTracking l'API n'est pas utilisée)
Suivi des publicités côté serveur	✓ Supporté : les balises se déclenchent en fonction des /v1/segment demandes utilisant l'identifiant de session	✓ Compatible (HLS uniquement) : utilise le balisage sans session avec des données de balise cryptées intégrées dans le via publicitaire. URIs #EXT-X-DEFINE:QUERYPARAM Nécessite HLS v11 ou version ultérieure. DASH n'est pas encore pris en charge.
Décoration Ad-ID	✓ Soutenu	✗ Non compatible

Détails de compatibilité

Prélecture des annonces

La prélecture des annonces n'est actuellement pas prise en charge.

Suppression des publicités

La suppression des publicités n'est pas prise en charge par les méthodes d'insertion de publicités guidées par le serveur, car les joueurs ne récupèrent que les publicités auxquelles ils veulent diffuser.

Comportement des publicités avant la diffusion

La synchronisation des annonces avant la diffusion fonctionne différemment selon les méthodes d'insertion :

- Insertion de publicités côté serveur : MediaTailor contrôle la diffusion des publicités en prédiffusion en fonction des paramètres de configuration
- Insertion d'annonces guidée par le serveur : MediaTailor insère des publicités prédiffusées en haut du manifeste. Votre lecteur diffuse d'abord ces publicités, puis lance votre contenu

Suivi des publicités

Le suivi côté client utilise différents mécanismes en fonction de la méthode d'insertion des annonces :

- Insertion de publicités côté serveur (SSAI) : utilise le point de terminaison de l'API `GetTracking`
- Insertion d'annonces guidée par le serveur (SGAI) : MediaTailor fournit des informations de suivi dans la `TRACKING` section de chaque réponse à la liste d'actifs. Le point de terminaison de l'`GetTrackingAPI` n'est pas utilisé. La réponse d'initialisation de session n'inclut pas `untrackingUrl`.

Le suivi côté serveur diffère également selon les méthodes :

- Insertion de publicités côté serveur (SSAI) : MediaTailor déclenche des balises lorsque le joueur extrait des segments publicitaires assemblés à l'aide de l'identifiant de session. `/v1/segment/`
- Insertion d'annonces guidée par le serveur (SGAI) : MediaTailor utilise le balisage sans session. MediaTailor intègre des données de balise cryptées (`awsBeaconData`, `awsBeaconDomain`, `awsConfigurationName`) dans le manifeste publicitaire URIs qu'il renvoie dans la liste des actifs. Le manifeste publicitaire utilise des `#EXT-X-DEFINE:QUERYPARAM` balises afin que le joueur remplace ces valeurs par segment URLs. Lorsque le joueur demande chaque segment publicitaire, MediaTailor déchiffre les données, déclenche la balise appropriée et redirige vers le segment de contenu. Lorsque le reporting côté serveur est actif, `TRACKING` cette section est MediaTailor omise dans la réponse de la liste des

actifs. Pour en savoir plus, consultez [Server-side suivi avec insertion d'annonces guidée par serveur \(SGAI\)](#).

Décoration Ad-ID

La décoration de l'identifiant publicitaire n'est pas compatible avec l'insertion d'annonces guidée par le serveur, car les champs qui remplissent X-AD-CREATIVE-SIGNALING les en-têtes ne sont connus que lorsque la liste des ressources est extraite, et non lorsque le manifeste est écrit.

MediaTailor configuration d'insertion publicitaire guidée par le serveur pour les diffusions en direct

AWS Elemental MediaTailor l'insertion de publicités guidée par le serveur pour le contenu en direct offre des avantages significatifs en termes de performances grâce à des manifestes pouvant être mis en cache. La configuration du SGAi pour le contenu en direct utilise les mêmes paramètres de base que la VOD, avec des considérations spécifiques relatives aux caractéristiques de diffusion en direct et au traitement en temps réel.

Exigences pour le SGAi en direct

Avant d'activer le SGAi pour le contenu en direct, assurez-vous que vous disposez des éléments suivants :

- Votre diffusion en direct inclut des marqueurs DATERANGE correctement formatés
- Les durées des pauses publicitaires sont cohérentes et prévisibles
- Votre CDN est configuré pour mettre en cache les manifestes SGAi de manière appropriée
- Les joueurs prennent en charge les workflows d'insertion de publicités guidés par le serveur
- Votre serveur de décision publicitaire peut traiter les demandes de contenu en direct en temps réel

Exigences relatives aux joueurs

Les joueurs doivent être configurés pour gérer correctement les manifestes en direct du SGAi :

- Support pour les flux de travail d'insertion de publicités guidés par le serveur
- Capacité à traiter les instructions d'insertion de publicités à partir des manifestes
- Gestion appropriée de la synchronisation et de la synchronisation des diffusions en direct

- Pour le contenu HLS : Support pour HLS version 8 et EXT-X-DATERANGE avec attribut CLASS. Version 11 pour le balisage côté serveur.
- Pour le contenu HLS : prise en charge de la substitution de EXT-X-DEFINE variables

Configuration de la lecture en direct

Pour activer le SGAI pour le contenu en direct, créez une configuration de lecture comportant les paramètres suivants :

Exemple Configuration de la lecture en direct SGAI

```
{
  "Name": "LiveSGAIConfig",
  "VideoContentSourceUrl": "https://your-live-origin.com/live/stream.m3u8",
  "AdDecisionServerUrl": "https://your-ads.com/ads",
  "PersonalizationThresholdSeconds": 1,
  "InsertionMode": "PLAYER_SELECT"
}
```

Voici les principales considérations à prendre en compte pour la configuration SGAI en direct :

VideoContentSourceUrl

Doit pointer vers un flux HLS en direct avec des marqueurs SCTE-35 DATERANGE correctement formatés. Le flux doit conserver des durées de segment et des variantes de débit cohérentes.

Demandes de manifeste SGAI en direct

Les manifestes dynamiques SGAI utilisent le même modèle d'URL que l'insertion publicitaire traditionnelle :

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/
manifest.m3u8?aws.insertionMode=GUIDED
```


Prélecture basée sur un manifeste pour SGAI en direct

Pour les flux de travail SGAI en direct, vous pouvez activer le battement de cœur par prélecture basé sur des manifestes afin de réduire le temps de latence des annonces publicitaires. Ajoutez `aws.guidedPrefetchMode=MANIFEST` à la demande de manifeste :

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/  
manifest.m3u8?aws.insertionMode=GUIDED&aws.guidedPrefetchMode=MANIFEST
```

Lorsque cette option MediaTailor est activée, ajoutez un identifiant de session (`aws.sessionId=<id>`) en tant que paramètre de requête à chaque URL du manifeste multimédia interstitiel (`/v1/i-media`) de la liste de lecture multivariante. Chaque fois que le lecteur actualise un manifeste i-media, la demande reçoit l'identifiant de session MediaTailor, qui MediaTailor sert à identifier la session et à mettre en file d'attente les demandes de prélecture pour les prochaines pauses publicitaires.

Important

Ne mettez pas en cache les manifestes i-media dans votre CDN lorsque vous utilisez le préchargement guidé. Le mécanisme du battement de cœur avant la lecture dépend de l'atteinte directe des demandes d'actualisation du manifeste du joueur. MediaTailor Si votre CDN met en cache et diffuse les `/v1/i-media` réponses, MediaTailor il ne reçoit pas les demandes de pulsation et ne peut pas déclencher le préchargement. Configurez votre CDN pour transmettre les `/v1/i-media/*` demandes au MediaTailor moment où il `aws.guidedPrefetchMode=MANIFEST` est utilisé.

La prélecture guidée est indépendante du mode de création de rapports. Que vous utilisiez le suivi côté serveur (par défaut) ou côté client (`aws.reportingMode=CLIENT`), les balises se déclenchent au moment de la diffusion, et non lorsque les publicités sont préluées. Pour obtenir des informations générales sur le fonctionnement de la prélecture des annonces MediaTailor, consultez [Publicités de prélecture](#)

Test de la configuration en direct de SGAI

Vérifiez votre configuration SGAI Live à l'aide des étapes de validation suivantes :

1. Génération de manifestes de test

Demandez l'URL du manifeste SGAI Live et vérifiez qu'il renvoie du contenu pouvant être mis en cache avec des instructions d'insertion d'annonces appropriées.

2. Vérifier la mise en cache du CDN

Vérifiez que votre CDN met en cache les manifestes SGAI conformément aux valeurs TTL configurées.

3. Tester l'insertion de publicités

Vérifiez que les joueurs peuvent insérer des publicités avec succès en suivant les instructions fournies dans les manifestes SGAI.

4. Surveiller les performances

Utilisez CloudWatch des métriques pour vérifier la réduction de la charge d'origine et l'amélioration des taux de réussite du cache.

MediaTailor configuration d'insertion publicitaire guidée par le serveur pour le contenu VOD

AWS Elemental MediaTailor l'insertion de publicités guidée par le serveur pour le contenu VOD offre des avantages significatifs en termes de performances grâce à des manifestes hautement cachables et à une réduction du traitement par le serveur. La configuration du SGAI pour le contenu VOD tire parti de la nature statique des video-on-demand ressources afin d'optimiser l'efficacité de la mise en cache et de minimiser les demandes d'origine, ce qui en fait la solution idéale pour les grandes bibliothèques de contenu présentant des modèles d'affichage répétés.

Exigences relatives à la VOD SGAI

Avant d'activer le SGAI pour le contenu VOD, assurez-vous que vous disposez des éléments suivants :

- Votre contenu VOD inclut des marqueurs publicitaires correctement formatés (SCTE-35 ou métadonnées chronométrées)
- Le contenu est stocké dans une source fiable avec une disponibilité constante
- Votre CDN est configuré pour mettre en cache les manifestes SGAI avec les valeurs TTL appropriées
- Les joueurs prennent en charge les flux de travail d'insertion publicitaire guidés par le serveur

- Votre serveur de décision publicitaire peut gérer les métadonnées et le ciblage spécifiques à la VOD

Exigences relatives aux joueurs

Les joueurs doivent être configurés pour gérer les manifestes VOD SGAI et l'insertion de publicités :

- Support pour les flux de travail d'insertion de publicités guidés par le serveur
- Capacité à traiter les conseils d'insertion de publicités à partir de manifestes de VOD
- Support pour l'insertion de publicités côté client pendant la lecture de VOD
- Gestion appropriée des opérations de recherche pendant les pauses publicitaires
- Support pour la durée du contenu et le suivi de position

Configuration de lecture VOD

Pour activer le SGAI pour le contenu VOD, créez une configuration de lecture comportant les paramètres suivants :

Exemple Configuration de lecture SGAI VOD

```
{
  "Name": "VODSGAIConfig",
  "VideoContentSourceUrl": "https://your-vod-origin.com/content/",
  "AdDecisionServerUrl": "https://your-ads.com/ads",
  "PersonalizationThresholdSeconds": 5,
  "InsertionMode": "PLAYER_SELECT"
}
```

Voici les principales considérations relatives à la configuration de la VOD SGAI :

VideoContentSourceUrl

Doit pointer vers votre bibliothèque de contenu VOD avec des modèles d'URL cohérents. Assurez-vous que l'origine peut gérer le volume de demandes attendu et assure une diffusion de contenu fiable.

ConfigurationAliases

Incluez des paramètres spécifiques à la VOD, tels que la durée du contenu, le genre ou les informations sur les séries, qui peuvent être utilisés pour le ciblage publicitaire sans affecter la capacité de mise en cache des manifestes.

ManifestProcessingRules

Activez le transfert des marqueurs publicitaires pour conserver les informations de synchronisation du contenu d'origine, ce qui est particulièrement important pour le contenu VOD dont les positions de pause publicitaire sont prédéfinies.

Demandes de manifeste SGAI VOD

Les manifestes VOD SGAI utilisent le même modèle d'URL que celui d'insertion d'annonces VOD traditionnelles.

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/content-path/manifest.m3u8?aws.insertionMode=GUIDED
```

Ciblage publicitaire spécifique à la VOD

Le contenu VOD offre des opportunités de ciblage publicitaire uniques :

Ciblage des métadonnées du contenu

Tirez parti des métadonnées du contenu VOD pour améliorer le ciblage publicitaire :

- Genre et catégorie : Cibler les publicités en fonction du type de contenu (série dramatique, comédie, documentaire)
- Évaluation du contenu : assurez-vous que le contenu publicitaire est adapté à l'âge (classifications G, PG, R)
- Séries et saisons : ciblez les publicités pour la continuité des séries ou le contenu connexe
- Date de sortie : objectif basé sur l'âge du contenu (nouvelles versions par rapport au contenu du catalogue)
- Durée du contenu : ajustez le chargement des annonces en fonction de la longueur totale du contenu

Affichage du ciblage contextuel

Les modèles de visionnage de la VOD permettent un ciblage publicitaire contextuel :

- Moment de la journée : ciblez les publicités en fonction du moment où le contenu est regardé
- Binge watching : ajustez la fréquence des publicités pour les utilisateurs qui regardent plusieurs épisodes
- Taux d'achèvement : cible basée sur les modèles d'achèvement du contenu historiques de l'utilisateur
- Type d'appareil : optimisez les formats publicitaires pour les appareils de visionnage (téléviseur, mobile, tablette)

Test de la configuration VOD SGAI

Vérifiez votre configuration SGAI VOD à l'aide des étapes de validation suivantes :

1. Génération de manifestes de test

Demandez un manifeste SGAI VOD URLs pour différents types de contenu et vérifiez qu'ils renvoient du contenu pouvant être mis en cache avec des conseils d'insertion d'annonces appropriés.

2. Vérifier la mise en cache du CDN

Vérifiez que votre CDN met en cache les manifestes SGAI conformément aux valeurs TTL configurées et qu'il atteint des taux de réussite élevés.

3. Tester l'insertion de publicités

Vérifiez que les joueurs peuvent insérer des publicités avec succès en suivant les instructions fournies dans les manifestes SGAI pour différents contenus VOD.

4. Opérations de recherche de tests

Vérifiez que la recherche dans le contenu VOD fonctionne correctement pendant les pauses publicitaires et que la position de lecture reste correcte.

5. Surveiller les performances

Utilisez CloudWatch des indicateurs pour vérifier la réduction de la charge d'origine, l'amélioration des taux de réussite du cache et les taux d'insertion de publicités réussis.

Principaux scénarios de test

Testez ces scénarios de VOD spécifiques :

- Contenu populaire : vérifiez les taux de réussite élevés du cache pour les ressources VOD fréquemment consultées
- Contenu de longue durée : testez l'insertion de publicités dans des films ou des épisodes longs comportant plusieurs interruptions publicitaires
- Contenu des séries : vérifiez la cohérence du ciblage publicitaire entre les épisodes d'une série
- Différents genres : testez le ciblage publicitaire en fonction des métadonnées du contenu et du genre

Bonnes pratiques d'optimisation de la VOD SGAI

Optimisez votre implémentation SGAI VOD pour des performances maximales :

Optimisation du cache

- Maximisez les valeurs TTL : utilisez des durées de cache plus longues pour les manifestes VOD, car le contenu ne change pas
- Minimiser les clés de cache : réduire les variations des clés de cache pour améliorer les taux de réussite
- Préchauffez le contenu populaire : mettez en cache le contenu VOD tendance ou en vedette
- Surveillez les performances du cache : suivez les taux de réussite du cache et optimisez-le en fonction des modèles d'utilisation

Optimisation de la diffusion de contenu

- Modèles d'URL cohérents : utilisez des structures d'URL prévisibles pour une meilleure mise en cache
- Standardisation des métadonnées : garantir la cohérence des métadonnées du contenu pour un ciblage publicitaire fiable
- Positionnement des pauses publicitaires : optimisez le placement des pauses publicitaires pour des transitions de contenu naturelles
- Variantes de qualité : assurez-vous que le SGAI fonctionne avec toutes les variantes de débit de votre contenu VOD

MediaTailor exigences relatives à l'intégration des serveurs publicitaires

Pour intégrer votre serveur publicitaire AWS Elemental MediaTailor, celui-ci doit envoyer du code XML conforme aux spécifications de l'IAB pour les versions prises en charge de VAST et VMAP. Vous pouvez utiliser un validateur VAST public pour vous assurer que les balises sont correctement formées.

AWS Elemental MediaTailor prend en charge les réponses VAST et VMAP des serveurs de décision publicitaire. AWS Elemental MediaTailor prend également en charge la transmission par proxy des métadonnées VPAID via notre API de reporting côté client pour l'insertion d'annonces côté client. Pour plus d'informations sur la création de rapports côté client, consultez [Client-side suivi des publicités](#).

MediaTailor prend en charge les versions suivantes de VAST, VMAP et VPAID :

- Jusqu'à [VAST 4.3](#)

MediaTailor accepte les versions de réponse jusqu'à VAST 4.3, mais certaines fonctionnalités avancées de VAST 4.0 et versions ultérieures ne sont pas prises en charge.

- [VMAP 1.0](#)
- [VPAID 2.0](#)

Des exigences VASTES

La réponse VAST de votre serveur de publicités doit contenir les éléments `TrackingEvents` conformes à IAB et les types d'événements standard, comme `impression`. Si vous incluez des événements de suivi non standard, si vous AWS Elemental MediaTailor rejetez la réponse VAST et ne diffusez pas d'annonce en faveur de cette fin.

VAST 3.0 a introduit la prise en charge des pods publicitaires, qui consiste en la livraison d'un ensemble de publicités linéaires séquentielles. Si une annonce spécifique dans un module publicitaire n'est pas disponible, AWS Elemental MediaTailor enregistre une erreur dans le journal des interactions de l'ADS. CloudWatch Ensuite, il tente d'insérer la prochaine ad dans le pod. De cette façon, MediaTailor parcourt les publicités du module jusqu'à ce qu'il en trouve une qu'il peut utiliser.

Ciblage

Pour cibler des joueurs spécifiques pour vos publicités, vous pouvez créer des modèles pour vos tags publicitaires et URLs. Pour de plus amples informations, veuillez consulter [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

AWS Elemental MediaTailor utilise par proxy le lecteur `user-agent` et `x-forwarded-for` les en-têtes lorsqu'il envoie la requête VAST au serveur publicitaire et lorsqu'il passe les appels de suivi côté serveur. Assurez-vous que votre serveur de publicités puisse gérer ces en-têtes. Sinon, vous pouvez utiliser `[session.user_agent]` ou `[session.client_ip]` et transmettre ces valeurs dans les chaînes de requête sur la balise et l'URL de la publicité. Pour de plus amples informations, veuillez consulter [MediaTailor variables de session pour les requêtes ADS](#).

Appels publicitaires

AWS Elemental MediaTailor appelle l'URL de vos annonces VAST telle que définie dans votre configuration. Il remplace tous les paramètres spécifiques au joueur ou à la session lors de l'appel publicitaire. MediaTailor suit jusqu'à sept niveaux de wrappers VAST et de redirections dans la réponse VAST. Dans les scénarios de diffusion en direct, MediaTailor passe des appels publicitaires simultanément au début de la diffusion de la publicité pour les joueurs connectés. En pratique, pour des raisons d'instabilité, ces appels peuvent être répartis sur quelques secondes. Assurez-vous que votre serveur publicitaire peut gérer le nombre de connexions simultanées requises par ce type d'appel. MediaTailor prend en charge la prélecture des réponses VAST pour les flux de travail en direct. Pour de plus amples informations, veuillez consulter [Publicités de prélecture](#).

Manipulation créative

Lorsqu'il AWS Elemental MediaTailor reçoit la réponse ADS VAST, il identifie pour chaque création le débit le plus élevé MediaFile pour le transcodage et l'utilise comme source. Il envoie ce fichier au on-the-fly transcodeur pour le transformer en rendus adaptés aux débits et résolutions des playlists multivariantes du lecteur. Pour de meilleurs résultats, assurez-vous que votre fichier multimédia au débit le plus élevé est un MP4 actif de haute qualité avec des préréglages de manifeste valides. Lorsque les préréglages du manifestes ne sont pas valides, les tâches de transcodage échouent et aucun résultat n'apparaît. Les exemples de préréglages non valides incluent les formats de fichiers d'entrée non pris en charge, tels que, et certaines spécifications de rendu ProRes, telles que la résolution 855 x 481.

Pour obtenir la liste des formats pris en charge pour les entrées de fichiers multimédia, consultez la MP4ligne [Formats d'entrée pris](#) en charge dans le guide de AWS Elemental MediaConvert l'utilisateur.

Indexation créative

AWS Elemental MediaTailor indexe de manière unique chaque création en fonction de la valeur de l'attribut fourni dans l'élément `<Creative>`. Si l'identifiant d'une création n'est pas spécifié, MediaTailor utilise l'URL du fichier multimédia pour l'index.

L'exemple de déclaration suivant affiche l'ID de la création.

```
<Creatives>
  <Creative id="57859154776" sequence="1">
```

Si vous définissez votre propre création IDs, utilisez un nouvel identifiant unique pour chaque création. Ne réutilisez pas vos créations IDs. AWS Elemental MediaTailor stocke le contenu créatif pour une utilisation répétée et trouve chacun grâce à son identifiant indexé. Lorsqu'une nouvelle création intervient, le service vérifie d'abord son ID par rapport à l'index. Si l'ID est présent, MediaTailor utilise le contenu stocké plutôt que de retraiter le contenu entrant. Si vous réutilisez un Creative ID, MediaTailor utilise l'ancienne annonce enregistrée et ne diffusez pas votre nouvelle annonce.

Extensions VAST fournies par des partenaires de diffusion d'annonces

Pour éviter les collisions avec la création IDs, vous pouvez utiliser les extensions fournies par les partenaires de diffusion de publicités pour la réponse VAST. MediaTailor prend en charge les extensions de SpringServe, Publica et FreeWheel Lorsque vous activez les remplacements d'extension VAST, MediaTailor remplace l'ID de création par défaut par la valeur de l'extension.

Pour activer cette fonctionnalité, [soumettez un ticket AWS Support](#) pour demander l'activation de la création basée sur l'extension IDs VAST. Incluez les informations suivantes dans le ticket de Support :

- AWS Région
- AWS ID de compte
- MediaTailor noms de configuration de lecture

Pour vérifier que les créations basées sur l'extension VAST IDs sont activées sur votre compte, nous vous recommandons également de demander l'activation de la RAW_ADS_RESPONSE journalisation sur une configuration de mise en scène ou de test de lecture. Grâce à la journalisation, vous pouvez consulter la réponse VAST originale reçue par l'ADS et confirmer que les bonnes créations IDs ont été utilisées.

Exigences relatives au VPAID

VPAID permet aux éditeurs de servir des vidéos publicitaires hautement interactives et de fournir des métriques de visibilité sur leurs flux monétisés. Pour plus d'informations sur VPAID, consultez la spécification [VPAID](#).

AWS Elemental MediaTailor prend en charge un mélange de publicités MP4 linéaires server-side-stitched VAST et de créations interactives client-side-inserted VPAID dans le même espace publicitaire. Il préserve l'ordre dans lequel ils apparaissent dans la réponse VAST. MediaTailor suit les redirections VPAID via un maximum de sept niveaux de wrappers. La réponse du reporting côté client inclut les métadonnées VPAID non wrappées.

Pour utiliser VPAID, suivez ces instructions :

- Configurez MP4 une liste pour vos créations VPAID. AWS Elemental MediaTailor remplit les espaces publicitaires VPAID avec votre liste configurée et fournit des métadonnées publicitaires VPAID que le joueur client peut utiliser pour diffuser les publicités interactives. Si vous n'avez pas de liste configurée, lorsqu'une publicité VPAID s'affiche, MediaTailor fournit les métadonnées publicitaires par le biais de rapports côté client comme à l'ordinaire. Il enregistre également une erreur CloudWatch concernant l'ardoise manquante. Pour plus d'informations, consultez [MediaTailor ardoise et insertion](#) et [Création d'une configuration MediaTailor de lecture](#).
- Utilisez les rapports côté client. AWS Elemental MediaTailor prend en charge VPAID via notre API de reporting côté client. Pour de plus amples informations, veuillez consulter [Client-side suivi des publicités](#).

En théorie, il est possible d'utiliser le reporting côté serveur par défaut avec VPAID. Cependant, si vous utilisez le reporting côté serveur, vous perdez toutes les informations sur la présence de la publicité VPAID et sur les métadonnées associées, car elles sont uniquement disponibles via l'API côté client.

- Dans les scénarios en direct, assurez-vous que la durée de validité de votre annonce, indiquée par `EXT-X-CUE-OUT: Duration`, est suffisamment longue pour tenir compte de l'interactivité des utilisateurs sur VPAID. Par exemple, si le code XML VAST spécifie une publicité VPAID d'une

durée de 30 secondes, pensez à configurer la durée de votre publicité pour qu'elle dure plus de 30 secondes. Ce délai supplémentaire donne aux utilisateurs plus de chances d'interagir avec l'annonce. Si vous n'ajoutez pas de temps, vous risquez de perdre les métadonnées VPAID car la durée restante de diffusion de la publicité n'est pas suffisante pour accueillir la publicité VPAID.

Buffet publicitaire

La spécification Interactive Advertising Bureau (IAB) Video Ad Serving Template (VAST) définit les pods publicitaires et les buffets d'annonces. Ce mécanisme prend en charge l'insertion ordonnée des annonces avec une solution de repli automatique en cas d'échec de l'insertion d'une annonce. Dans un module publicitaire, les publicités séquencées sont diffusées dans l'ordre. Les publicités autonomes constituent un buffet à partir duquel les annonces de remplacement MediaTailor sont sélectionnées en cas d'échec d'une annonce séquencée.

AWS Elemental MediaTailor prend en charge cette spécification grâce au `AdSequencingMode` réglage `AdDecisionServerConfiguration` du `PlaybackConfiguration`. Lorsque cette option est définie sur `FOLLOW_AD_SEQUENCE`, MediaTailor insère les annonces séquencées dans l'ordre et utilise des publicités autonomes uniquement comme substituts lorsqu'aucune annonce séquencée n'est insérée.

Exemple de réponse VAST

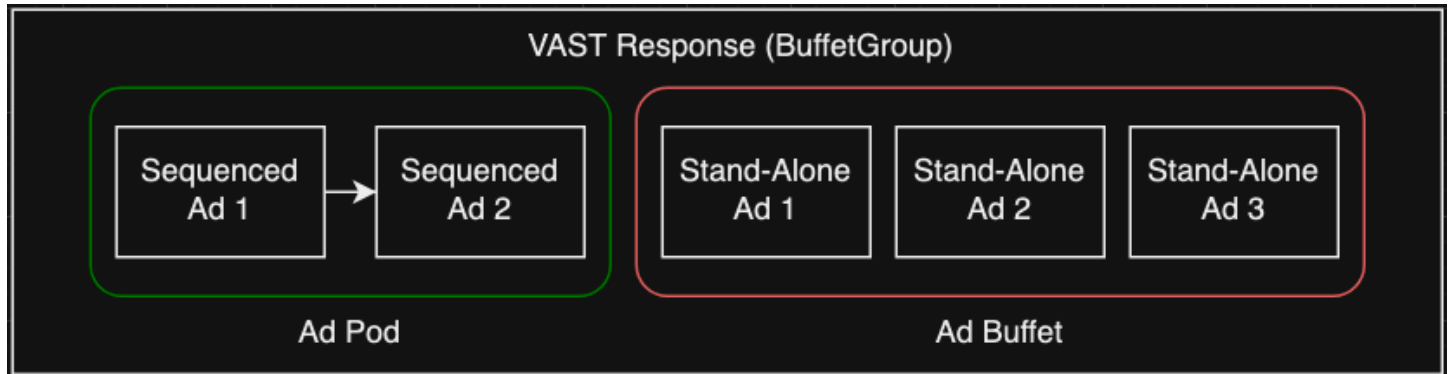
L'exemple suivant montre une réponse VAST qui contient des publicités séquencées et des publicités autonomes :

```
<VAST>
  <Ad sequence="1" id="sequencedAd1">...</Ad>      <!-- Sequenced Ad 1 -->
  <Ad sequence="2" id="sequencedAd2">...</Ad>      <!-- Sequenced Ad 2 -->
  <Ad id="standAloneAd1">...</Ad>                  <!-- Standalone Ad 1 -->
  <Ad id="standAloneAd2">...</Ad>                  <!-- Standalone Ad 2 -->
  <Ad id="standAloneAd3">...</Ad>                  <!-- Standalone Ad 3 -->
</VAST>
```

Dans cet exemple, les deux publicités séquencées forment un module publicitaire. Les trois publicités indépendantes constituent le buffet publicitaire. Ensemble, ils constituent un groupe de buffets. En cas d'échec de `sequencedAd1`, essayez d'abord de la remplacer par une annonce autonome éligible provenant du même groupe de buffets. Si aucune annonce autonome éligible n'est

disponible dans ce groupe, MediaTailor sélectionnez parmi les annonces autonomes des autres groupes de buffets.

Le schéma suivant montre une réponse VAST avec un module publicitaire (annonces séquencées) et un buffet d'annonces (annonces autonomes) qui forment ensemble un groupe de buffets.



Concepts clés

Les termes suivants décrivent les concepts de base d'un buffet publicitaire :

- **Publicités séquencées** : annonces dont l'<Ad>élément contient un `sequence` attribut. Ils forment un ad pod ordonné et sont insérés dans un ordre séquentiel croissant.
- **Publicités autonomes (sous forme de buffet)** : annonces sans `sequence` attribut. Ils ne sont utilisés que comme substituts lorsqu'une annonce séquencée ne parvient pas à être insérée.
- **Buffer group** : MediaTailor associe un module publicitaire à ses publicités autonomes au sein d'une seule réponse VAST. Lorsqu'une annonce séquencée est un emballage, son VAST redirigé forme son propre groupe de buffets indépendant. Lorsque vous remplacez une annonce séquencée qui a échoué, MediaTailor donne la priorité aux publicités autonomes du même groupe de buffets. Si aucune annonce autonome éligible n'existe dans le même groupe de buffets, MediaTailor revient aux annonces autonomes provenant d'autres groupes de buffets.

Note

`sequence` fait référence à l'attribut <Ad> au niveau de l'élément, et non à l'`sequence` attribut trouvé dans l'<Creative>élément.

Les avantages du buffet publicitaire

Sans buffet publicitaire, MediaTailor traite toutes les publicités de la même manière, ce qui peut entraîner les problèmes suivants :

- VOD — Les publicités autonomes sont insérées même lorsqu'aucun échec ne se produit, ce qui perturbe les positions des pauses publicitaires convenues et les surcharge.
- En direct : les publicités Lower-revenue autonomes peuvent être insérées avant les publicités séquencées et occuper toute la durée de validité, ce qui ne laisse aucune place aux publicités séquencées à revenus plus élevés.

Lorsque le buffet d'annonces est activé, n' MediaTailor insère que des publicités séquencées et utilise des publicités autonomes uniquement comme substituts lorsqu'une annonce séquencée ne parvient pas à être insérée. Cela garantit que les publicités les plus rentables sont diffusées en premier.

Warning

Si la réponse de votre serveur de décision publicitaire (ADS) ne contient aucune annonce séquencée (aucun <Ad> élément ne possède d'sequenceattribut), l'activation d'un buffet publicitaire entraîne l'absence d'insertion d'annonce.

Activation du buffet publicitaire

Ad Buffet est une fonctionnalité optionnelle que vous configurez pour chaque configuration de lecture à l'aide de l'PutPlaybackConfigurationAPI.

AdSequencingMode valeurs

Le tableau suivant décrit les AdSequencingMode valeurs disponibles.

Value	Description
IGNORE_AD_SEQUENCE (par défaut)	MediaTailor insère les publicités dans l'ordre dans lequel elles apparaissent dans la réponse VAST, quels que soient les attributs de séquence.

Value	Description
FOLLOW_AD_SEQUENCE	MediaTailor insère des publicités séquencées dans l'ordre pour les flux de travail en direct et en VOD. Les publicités séquencées qui ont échoué sont remplacées par des publicités autonomes.
FOLLOW_AD_SEQUENCE_ONLY_LIVE	Le comportement des buffets publicitaires est activé pour les flux de travail en direct uniquement.
FOLLOW_AD_SEQUENCE_ONLY_VOD	Le comportement des buffets publicitaires est activé pour les flux de vidéo à la demande uniquement.

Note

Lorsque vous activez le buffet publicitaire, MediaTailor respecte les `Wrapper.allowMultipleAds` attributs `Wrapper.fallbackOnNoAd` et.

Configuration de Midroll

L'exemple suivant active le buffet publicitaire pour les flux de travail en direct et en VOD :

```
{
  "PlaybackConfiguration": {
    "AdDecisionServerConfiguration": {
      "VastResponse": {
        "AdSequencingMode": "FOLLOW_AD_SEQUENCE"
      }
    }
  }
}
```

L'exemple suivant active le buffet publicitaire pour les flux de travail en direct uniquement :

```
{
  "PlaybackConfiguration": {
    "AdDecisionServerConfiguration": {
      "VastResponse": {
        "AdSequencingMode": "FOLLOW_AD_SEQUENCE_ONLY_LIVE"
      }
    }
  }
}
```

```

    }
  }
}

```

L'exemple suivant active le buffet publicitaire pour les flux de travail de VOD uniquement :

```

{
  "PlaybackConfiguration": {
    "AdDecisionServerConfiguration": {
      "VastResponse": {
        "AdSequencingMode": "FOLLOW_AD_SEQUENCE_ONLY_VOD"
      }
    }
  }
}

```

Configuration du pré-roll en direct

L'exemple suivant active le buffet publicitaire pour les annonces en direct avant le lancement :

```

{
  "PlaybackConfiguration": {
    "LivePrerollConfiguration": {
      "AdDecisionServerConfiguration": {
        "VastResponse": {
          "AdSequencingMode": "FOLLOW_AD_SEQUENCE"
        }
      }
    }
  }
}

```

Note

LivePrerollConfiguration prend en charge que FOLLOW_AD_SEQUENCE et IGNORE_AD_SEQUENCE. Configurez le buffet publicitaire pour le préroll de la VOD via la configuration midroll.

Comment fonctionne le buffet publicitaire

Conditions de remplacement des annonces

Si une annonce séquencée rencontre l'un des cas d'échec suivants, MediaTailor tente de remplacer l'annonce séquencée par une annonce autonome éligible du même groupe de buffets. Si aucune annonce autonome n'est disponible dans le même groupe de buffets, MediaTailor utilise une annonce autonome provenant d'autres groupes de buffets, si disponible.

Cas de défaillance	Description
Échec de l'analyse VAST	Une annonce séquencée ne parvient pas à être analysée en raison de l'absence d'éléments VAST requis (par exemple <code>MediaFile s Creatives</code> ,, ou <code>Impression</code> sont manquants <code>MediaFile</code> ou <code>VASTAdTagURI</code> non valides).
Défaillance de la résolution du wrapper VAST	Une redirection expire, échoue ou renvoie une réponse vide.
Annonce VPAID supprimée	Une annonce séquencée est VPAID et la session utilise des rapports côté serveur ou aucune liste n'est configurée.
Le transcodage publicitaire n'est pas prêt	Le statut de transcodage d'une annonce séquencée n'est pas SUCCESS (par exemple, IN_PROGRESS ou ERROR).
Durée de disponibilité dépassée	La durée d'une annonce séquencée dépasse la durée de disponibilité restante (flux de travail en direct).

Éligibilité aux publicités autonomes

Pour être éligible en tant que remplacement, une annonce autonome doit répondre aux critères suivants :

- Ne soyez pas une annonce VPAID
- Avoir terminé le transcodage
- Avoir une durée qui correspond à la durée de disponibilité restante

Sélection d'annonces autonome

Lorsqu'une annonce séquencée échoue, MediaTailor sélectionne une annonce de remplacement parmi les annonces autonomes éligibles en utilisant la priorité suivante :

1. D'abord, même groupe de buffets : MediaTailor préfère les publicités indépendantes issues du groupe de buffets de l'annonce qui a échoué.
2. Autres groupes de buffets — Utilisés uniquement s'il n'existe aucune annonce éligible dans le même groupe de buffets.

Si aucune annonce autonome éligible n'est trouvée, le poste n'est pas pourvu et la publicité appropriée `SkippedReason` est émise.

Comportement du wrapper VAST

La gestion d'une réponse de redirection Wrapper par MediaTailor dépend de la valeur de deux variables Wrapper : `Wrapper.fallbackOnNoAd` et `Wrapper.allowMultipleAds`

- `fallbackOnNoAd` prend effet lorsque la réponse de redirection est vide, expire ou qu'une erreur se produit lors de la demande ADS.
- `allowMultipleAds` prend effet si la réponse ADS contient une ou plusieurs annonces.

VAST vide, erreur de demande ou délai d'expiration de la demande

Le tableau suivant décrit le MediaTailor comportement lorsque la réponse de redirection est vide, expire ou qu'une erreur se produit.

repli OnNoAd	MediaTailor comportement
true (par défaut)	MediaTailor sélectionne une annonce autonome pour remplacer le wrapper défaillant, en préférant le même groupe de buffets. Si aucune annonce éligible pour le même groupe n'est disponible, choisissez MediaTailor parmi les autres groupes de buffets.
false	MediaTailor supprime la position de l'annonce et passe à l'annonce suivante dans le module. Aucun remplacement n'a lieu.

Exemple fallback OnNoAd =true (échec de la redirection)

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad sequence="2">
    <Wrapper fallbackOnNoAd="true">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper : VAST vide, délais d'expiration ou erreurs.

VAST résolu :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <!-- standAloneAd1 replaced failed Wrapper Ad and now has the Wrapper's sequence (2) -->
  <Ad sequence="2" id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Ordre de lecture :

1. Série D1
2. stand AloneAd 1

Exemple fallback OnNoAd =false (la redirection échoue)

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad sequence="2">
    <Wrapper fallbackOnNoAd="false">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

```
</Wrapper>
</Ad>
<Ad id="standAloneAd1">...</Ad>
<Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper : VAST vide, délais d'expiration ou erreurs.

VAST résolu :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <!-- sequence 2 is not replaced -->
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Ordre de lecture :

1. Série D1

La réponse contient une ou plusieurs annonces

Le tableau suivant décrit le MediaTailor comportement lorsque la réponse de redirection contient une ou plusieurs publicités.

autoriser MultipleAds	MediaTailor comportement
false (par défaut)	MediaTailor sélectionne une annonce dans la réponse de redirection et supprime les annonces restantes. L'annonce sélectionnée hérite de la valeur de séquence de l'annonce Wrapper Ad parent. Stratégie de sélection : MediaTailor sélectionne l'annonce dont la séquence est la plus basse dans la réponse de redirection. Si aucune annonce séquencée n'est disponible, choisissez MediaTailor une annonce autonome, si elle est disponible.
true	Lorsque le wrapper possède un attribut de séquence, MediaTailor insère des publicités de redirection à la position du wrapper. Lorsque le wrapper n'a aucun attribut de séquence, MediaTailor insère des

autoriser MultipleAds	MediaTailor comportement
	publicités de redirection après les annonces séquencées du parent. Les publicités résolues constituent leur propre groupe de buffets.

Exemple autoriser MultipleAds = faux, le wrapper a une séquence = 2

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad sequence="2">
    <Wrapper allowMultipleAds="false">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper :

```
<VAST>
  <Ad sequence="1" id="redirectSeqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd2">...</Ad>
  <Ad id="redirectStandAloneAd1">...</Ad>
</VAST>
```

VAST résolu :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd1">...</Ad> <!-- redirectSeqAd1 selected, inherits
Wrapper's sequence (2) -->
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Ordre de lecture :

1. Série D1
2. rediriger SeqAd 1
3. Série D3

Exemple allow MultipleAds =false, le wrapper n'a pas de séquence

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad>
    <Wrapper allowMultipleAds="false">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper :

```
<VAST>
  <Ad sequence="1" id="redirectSeqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd2">...</Ad>
  <Ad id="redirectStandAloneAd1">...</Ad>
</VAST>
```

VAST résolu :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad id="redirectSeqAd1">...</Ad> <!-- redirectSeqAd1 selected, inherits Wrapper's
sequence (UNDEFINED) -->
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Ordre de lecture :

1. Série D1
2. Série D3

 Note

redirect SeqAd 1 n'est pas joué car aucune séquence n'est définie.

Exemple autoriser MultipleAds = vrai, le wrapper a une séquence = 2

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad sequence="2">
    <Wrapper allowMultipleAds="true">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper :

```
<VAST>
  <Ad sequence="1" id="redirectSeqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd2">...</Ad>
  <Ad id="redirectStandAloneAd1">...</Ad>
</VAST>
```

VAST résolu :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <!-- Redirect response merged at sequence 2 -->
  <Ad sequence="1" id="redirectSeqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd2">...</Ad>
  <Ad id="redirectStandAloneAd1">...</Ad>
```

```
<Ad sequence="3" id="seqAd3">...</Ad>
<Ad id="standAloneAd1">...</Ad>
<Ad id="standAloneAd2">...</Ad>
</VAST>
```

Ordre de lecture :

1. Série D1
2. rediriger SeqAd 1
3. redirection SeqAd 2
4. Série D3

Exemple allow MultipleAds =true, le wrapper n'a pas de séquence

Parent VAST :

```
<VAST>
  <Ad sequence="1" id="seqAd1">...</Ad>
  <Ad>
    <Wrapper allowMultipleAds="true">
      <VASTAdTagURI>https://ads.example.com/vast</VASTAdTagURI>
    </Wrapper>
  </Ad>
  <Ad sequence="3" id="seqAd3">...</Ad>
  <Ad id="standAloneAd1">...</Ad>
  <Ad id="standAloneAd2">...</Ad>
</VAST>
```

Réponse de redirection du wrapper :

```
<VAST>
  <Ad sequence="1" id="redirectSeqAd1">...</Ad>
  <Ad sequence="2" id="redirectSeqAd2">...</Ad>
  <Ad id="redirectStandAloneAd1">...</Ad>
</VAST>
```

VAST résolu :

```
<VAST>
```

```

<Ad sequence="1" id="seqAd1">...</Ad>
<Ad sequence="3" id="seqAd3">...</Ad>
<!-- Redirect response merged after sequenced ads -->
<Ad sequence="1" id="redirectSeqAd1">...</Ad>
<Ad sequence="2" id="redirectSeqAd2">...</Ad>
<Ad id="redirectStandAloneAd1">...</Ad>
<Ad id="standAloneAd1">...</Ad>
<Ad id="standAloneAd2">...</Ad>
</VAST>

```

Ordre de lecture :

1. Série D1
2. Série D3
3. rediriger SeqAd 1
4. redirection SeqAd 2

Surveillance des métriques et des journaux

Modifications apportées aux indicateurs existants

Lorsque le buffet publicitaire est activé, les mesures existantes suivantes modifient le comportement.

Métrique	Comportement avec un buffet publicitaire
AdDecisionServer.Ads	Nombre de publicités séquencées analysées à partir de la réponse ADS.
AdDecisionServer.Duration	Durée totale des publicités séquencées uniquement.
AdDecisionServer.FillRate	Taux de remplissage calculé à l'aide de publicités séquencées uniquement.

Nouvelles métriques

Les nouvelles statistiques suivantes sont disponibles lorsque le buffet publicitaire est activé.

Métrique	Description
<code>AdNotReady.SequencedAd</code>	Nombre de publicités séquencées qui ne sont pas prêtes à être insérées.
<code>AdNotReady.StandaloneAd</code>	Nombre d'annonces autonomes qui ne sont pas prêtes à être insérées (à l'exclusion des annonces ignorées <code>StandaloneAdNotNeeded</code>).
<code>Avail.FillRate.SequencedAd</code>	Pourcentage d'audience généré par des publicités séquencées.
<code>Avail.FillRate.StandaloneAd</code>	Pourcentage de diffusion généré par des publicités indépendantes.
<code>Avail.FilledDuration.SequencedAd</code>	Durée d'utilisation remplie de publicités séquencées.
<code>Avail.FilledDuration.StandaloneAd</code>	Durée de disponibilité remplie d'annonces autonomes.
<code>AdsBilled.SequenceAd</code>	Nombre d'annonces facturées qui sont séquencées.
<code>AdsBilled.StandaloneAd</code>	Nombre de publicités facturées indépendantes.
<code>SkippedReason.StandaloneAdNotNeeded</code>	Émis lorsqu'une annonce autonome n'était pas nécessaire car toutes les publicités séquencées ont été insérées avec succès.

Nouvelle raison pour laquelle l'annonce a été ignorée

La nouvelle raison suivante pour laquelle l'annonce a été ignorée est disponible lorsque le buffet publicitaire est activé.

Raison	Description
STANDALONE_AD_NOT_NEEDED	Une annonce autonome était éligible mais n'était pas nécessaire car toutes les publicités séquencées ont été insérées avec succès. Aucune balise d'erreur n'est déclenchée.

Pour les publicités autonomes qui ne peuvent pas être insérées (par exemple, VPAID ou Transcode not ready), la raison réelle de l'échec est émise (par exemple, TRANSCODE_IN_PROGRESS ou VPAID_AD) sous la forme. `SkippedReason`

Beaconing d'erreur

MediaTailor gère les balises d'erreur pour les buffets publicitaires comme suit :

- Publicités séquencées qui échouent : des balises d'erreur sont déclenchées (comportement existant).
- Publicités autonomes qui ne sont pas prêtes (l'état du transcodage des publicités n'est pas une réussite) : des balises d'erreur sont déclenchées (comme pour les publicités séquencées).
- Publicités autonomes non utilisées qui pouvaient être insérées mais qui n'étaient pas nécessaires : aucune balise d'erreur n'est déclenchée.

Changements apportés aux événements ADS

Lorsque le buffet publicitaire est activé, MediaTailor ajoute les champs suivants à deux événements ADS :

Événement VAST_RESPONSE

Chacun VastAd des champs de l'VAST_RESPONSE événement inclut les champs suivants.

Champ	Type	Description
buffetGroup	Chaîne	MediaTailor chaîne générée qui définit le groupe de buffets auquel appartient l'annonce.
sequence	Int	La valeur de séquence de l'Adélément.

Événement FILLED_AVAIL

Chaque entrée dans `creativeAds` et `skippedAds` dans l'`FILLED_AVAIL` événement comprend les champs suivants.

Champ	Type	Description
<code>buffetGroup</code>	Chaîne	MediaTailor chaîne générée qui définit le groupe de buffets auquel appartient l'annonce.
<code>sequence</code>	Int	La valeur de séquence de l'Adélément.

Ces champs vous permettent de déterminer :

- Quelles publicités ont été analysées comme séquencées par rapport à des publicités autonomes
- Quelles annonces ont été insérées et leur type
- Pourquoi une annonce séquencée a-t-elle été ignorée et a-t-elle été remplacée par une annonce autonome ?
- Pourquoi une annonce indépendante n'était pas disponible

Consultez aussi

Pour plus d'informations sur l'`PutPlaybackConfigurationAPI` et la spécification IAB VAST, consultez les rubriques suivantes :

- [PutPlaybackConfiguration Référence API](#) : référence complète des paramètres pour la configuration `AdSequencingMode` et les paramètres associés.
- [IAB VAST 4.x Specification](#) sur le IAB Tech Lab site Web — La norme du secteur définissant les pods publicitaires, les publicités séquencées et les publicités autonomes.
- [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#) — Référence complète pour toutes les MediaTailor CloudWatch statistiques, y compris les nouvelles statistiques des buffets publicitaires.

MediaTailor gestion de la configuration de lecture

Cette section décrit les principales tâches relatives à la gestion des configurations AWS Elemental MediaTailor de lecture. Vous pouvez apprendre à créer une nouvelle configuration pour configurer des flux de contenu et fournir un accès aux appareils de lecture, afficher les détails d'une configuration existante, modifier une configuration pour mettre à jour des paramètres tels que les serveurs d'origine et les serveurs de décision publicitaire, et supprimer une configuration qui n'est plus nécessaire.

Rubriques

- [Création d'une configuration MediaTailor de lecture](#)
- [Affichage des détails MediaTailor de configuration](#)
- [Modification des paramètres MediaTailor de configuration](#)
- [Supprimer des MediaTailor configurations](#)

Création d'une configuration MediaTailor de lecture

Cette rubrique explique comment créer une configuration pour commencer à recevoir des flux de contenu AWS Elemental MediaTailor. Il montre également comment fournir un point d'accès aux périphériques de lecture en aval pour demander du contenu.

Vous pouvez utiliser la AWS Elemental MediaTailor console, le AWS Command Line Interface (AWS CLI) > ou l' MediaTailor API pour créer une configuration. Pour plus d'informations sur la création d'une configuration via l' MediaTailor API AWS CLI or, consultez la référence de l'[AWS Elemental MediaTailor API](#) .

Lorsque vous créez une configuration, ne saisissez pas d'informations d'identification sensibles dans des champs de forme libre tels que le champ Nom de la configuration. Les informations d'identification peuvent inclure des éléments tels que les numéros de compte client. En outre, n'utilisez pas d'informations d'identification lorsque vous travaillez dans la MediaTailor console, l'API REST AWS CLI, les kits de AWS développement logiciel (SDK). Toutes les données que vous saisissez MediaTailor peuvent être récupérées pour être incluses dans les journaux de diagnostic ou Amazon CloudWatch Events.

Pour ajouter une configuration (console)

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.

2. Sur la page Configurations, choisissez Create configuration (Créer une configuration).
3. Renseignez les champs concernant la configuration, comme décrit dans les rubriques suivantes :
 - [Réglages requis](#)
 - [Paramètres de configuration facultatifs](#)

4. Choisissez Créer une configuration.

AWS Elemental MediaTailor affiche la nouvelle configuration dans le tableau de la page Configurations.

5. (Recommandé) Configurez un CDN AWS Elemental MediaTailor pour les demandes de manifeste et de reporting. Vous pouvez utiliser les URL de lecture de la configuration pour la configuration du CDN. Pour plus d'informations sur la configuration d'un réseau de diffusion de contenu (CDN) pour les demandes de manifeste et de reporting, consultez [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#).

Réglages requis

Lorsque vous créez une configuration, vous devez inclure les paramètres obligatoires suivants.

Nom

Saisissez un nom unique qui décrit la configuration. Le nom est l'identifiant principal de la configuration. La longueur maximale autorisée est de 512 caractères.

Source de contenu

Saisissez le préfixe d'URL pour le manifeste de ce flux, sans l'ID d'asset. La longueur maximale est de 512 caractères.

Par exemple, le préfixe d'URL `http://origin-server.com/a/` est valide pour une URL de playlist multivariante HLS de `http://origin-server.com/a/main.m3u8` et pour une URL DASH MPD de `http://origin-server.com/a/dash.mpd`. Vous pouvez également saisir un préfixe plus court tel que `http://origin-server.com`, mais le `/a/` doit être inclus dans l'ID de ressource de la demande de contenu du lecteur.

Note

Si l'origine de votre contenu utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. Il ne peut pas s'agir d'un certificat auto-signé. Si vous utilisez un

certificat auto-signé, vous AWS Elemental MediaTailor ne parvenez pas à vous connecter à l'origine du contenu et ne pouvez pas diffuser de manifestes en réponse aux demandes des joueurs.

Serveur de décision publicitaire

Saisissez l'URL de votre serveur ADS. Il s'agit de l'URL avec les variables comme décrit dans [Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête](#) ou l'URL VAST statique que vous utilisez à des fins de test. La longueur maximale est de 25 000 caractères.

Note

Si votre ADS utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. Il ne peut pas s'agir d'un certificat auto-signé. La même chose s'applique également aux URL des publicités au format mezzanine renvoyées par l'ADS. Si vous utilisez un certificat auto-signé, vous ne pouvez pas récupérer et intégrer des publicités dans les manifestes à partir de l'origine du contenu.

Paramètres de configuration facultatifs

Vous pouvez éventuellement configurer des alias de configuration, des détails de personnalisation et des paramètres avancés dans la MediaTailor console, l'API MediaTailor ou le AWS Command Line Interface (AWS CLI).

Alias de configuration

Vous trouverez ci-dessous des alias de configuration facultatifs que vous pouvez configurer dans la MediaTailor console ou à l'aide de l'API MediaTailor.

Variable de paramètre du joueur

Pour une configuration dynamique du domaine lors de l'initialisation de la session, ajoutez une ou plusieurs variables de paramètres de lecteur.

Pour plus d'informations sur l'utilisation des variables de paramètres du lecteur pour configurer dynamiquement des domaines, consultez [MediaTailor variables de domaine pour plusieurs sources de contenu](#).

Configuration du journal

Les paramètres de configuration des journaux sont les suivants.

Pourcentage activé

Définit le pourcentage des journaux de session de configuration de lecture qui sont MediaTailor écrits dans les CloudWatch journaux. Par exemple, si votre configuration de lecture comporte 1 000 sessions et que vous définissez le pourcentage activé sur 60, MediaTailor écrit 600 journaux de session dans CloudWatch Logs.

Lorsque vous activez cette option, un rôle lié à un service est MediaTailor automatiquement créé qui permet d'écrire et MediaTailor de gérer les journaux de session dans votre compte CloudWatch Logs. Pour de plus amples informations, veuillez consulter [Utilisation de rôles liés à des services pour MediaTailor](#).

Stratégies de journalisation

Indique la méthode utilisée pour collecter les journaux MediaTailor émis. Pour envoyer les journaux directement à CloudWatch Logs, choisissez `LEGACY_CLOUDWATCH`. Pour envoyer des CloudWatch journaux à Logs, qui les envoie ensuite à la destination de votre choix, choisissez `VENDED_LOGS`. Les destinations prises en charge sont un groupe de CloudWatch journaux Logs, un compartiment Amazon S3 et un flux Amazon Data Firehose.

Une configuration supplémentaire est requise pour les journaux vendus. Pour la configuration, voir [Utilisation de journaux vendus](#).

Événements d'inscription au journal des interactions ADS

Indique qui MediaTailor émettra `RAW_ADS_RESPONSE` des journaux pour les sessions initialisées avec cette configuration.

L'événement de `RAW_ADS_RESPONSE` journal contient l'intégralité de la réponse VAST ou VMAP de l'ADS. En tant que tels, les journaux peuvent être volumineux et peuvent augmenter vos coûts d'enregistrement.

Exclusion d'événements dans le journal des interactions ADS

Indique que les événements sélectionnés MediaTailor n'apparaîtront pas dans les journaux qui décrivent les interactions avec l'ADS.

Pour une description des événements du journal ADS, consultez [Journaux ADS](#).

Exclusion d'événements dans le journal des interactions du service Manifest

Indique que les événements sélectionnés MediaTailor n'apparaîtront pas dans les journaux qui décrivent les interactions avec le service de manifeste.

Pour une description des événements du journal de service du manifeste, consultez [Journaux des manifestes](#).

Conditionnement publicitaire

Vous trouverez ci-dessous les mesures à prendre MediaTailor pour conditionner les publicités avant de les intégrer à un flux de contenu.

Conditionnement de fichiers multimédia en streaming

Détermine la logique MediaTailor à utiliser pour choisir les publicités à diffuser.

- Lorsque le conditionnement des fichiers multimédia en streaming est réglé sur `Transcoder`, les fichiers multimédia MediaTailor sont transcodés lors de la progressive livraison et les insèrent dans le manifeste. S'il n'y a pas assez de publicités contenant des fichiers multimédia pour les diffuser, les MediaTailor transcode et les utilise avec progressive streaming diffusion.
- Lorsque le conditionnement des fichiers multimédia en streaming est réglé sur `Aucun`, les MediaTailor publicités contenant des fichiers multimédia streaming de diffusion sont intégrées au manifeste sans les transcoder. S'il n'y a pas assez de publicités contenant des fichiers multimédia pour les diffuser, les MediaTailor transcode et les utilise avec streaming progressive diffusion.

Détails de personnalisation

Vous trouverez ci-dessous les détails de personnalisation que vous pouvez configurer dans la MediaTailor console ou à l'aide de l' MediaTailor API.

Slate ad (Liste de publicités)

Entrez l'URL d'une ressource MP4 de haute qualité MediaTailor utilisée pour remplir le temps d'affichage des annonces. La configuration de Slate est facultative pour la plupart des flux de travail, mais obligatoire pour les publicités VPAID. Pour des informations complètes sur le comportement de l'ardoise, les exigences de configuration et le moment où la ardoise est affichée, consultez [MediaTailor ardoise et insertion](#).

Pare-chocs de démarrage

URL de l'emplacement de l'actif Start Bumper. Les bumpers sont de courts clips vidéo ou audio diffusés au début ou à la fin d'une pause publicitaire. Ils peuvent être stockés sur le S3 d'Amazon ou sur un autre service de stockage. Pour en savoir plus sur les pare-chocs, voir [MediaTailor pare-chocs et insertion](#).

Pare-chocs d'extrémité

URL de l'emplacement de l'actif End Bumper. Les bumpers sont de courts clips vidéo ou audio diffusés au début ou à la fin d'une pause publicitaire. Ils peuvent être stockés sur le S3 d'Amazon ou sur un autre service de stockage. Pour en savoir plus sur les pare-chocs, voir [MediaTailor pare-chocs et insertion](#).

Seuil de personnalisation

Ne s'applique que lorsqu'il est utilisé conjointement avec Slate Ad. Définit la durée maximale d'une annonce sous-remplie (en secondes) autorisée pendant une pause publicitaire avant l'abandon de la personnalisation et l'affichage du contenu sous-jacent. Cette fonctionnalité s'applique au remplacement de publicités dans les flux en direct et en VOD, plutôt qu'à l'insertion de publicités, car elle repose sur un flux de contenu sous-jacent. Pour des scénarios de comportement détaillés et des informations complètes, consultez [Comment fonctionne le seuil de personnalisation](#).

Serveur ADS de preroll en direct

Pour insérer des publicités au début d'un flux en direct avant que le contenu principal soit lancé, saisissez l'URL du preroll à partir du serveur ADS. Il s'agit de l'URL avec les variables comme décrit dans [Étape 3 : Configuration de l'URL de la demande ADS et des paramètres de requête](#) ou l'URL VAST statique que vous utilisez à des fins de test. La longueur maximale est de 25 000 caractères.

Note

Si votre ADS utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. Il ne peut pas s'agir d'un certificat auto-signé. La même chose s'applique également aux URL des publicités au format mezzanine renvoyées par l'ADS. Si vous utilisez un certificat auto-signé, vous ne pouvez pas récupérer et intégrer des publicités dans les manifestes à partir de l'origine du contenu.

Pour plus d'informations sur le fonctionnement du pré-déploiement (preroll), consultez [MediaTailor pré-enroulement et insertion](#).

Durée maximale autorisée pour la diffusion de preroll

Lorsque vous insérez des publicités au début d'un flux en direct, entrez la durée maximale autorisée pour la diffusion de publicités preroll. MediaTailor ne dépassera pas cette durée lors de l'insertion de publicités. Si la réponse de l'ADS contient plus d'annonces que ce qui correspond à cette durée, MediaTailor elle propose autant d'annonces que possible, sans dépasser la durée. Pour plus de détails sur la façon dont MediaTailor les remplissages sont disponibles, consultez [Comportement d'assemblage des publicités en direct](#).

Utiliser le mode de suppression

Définit le mode de suppression de la diffusion, également connu sous le nom de suppression des publicités. Par défaut, la suppression des publicités est désactivée et MediaTailor toutes sont remplies d'annonces ou de listes. Lorsque le mode est défini sur `BEHIND_LIVE_EDGE`, la suppression des publicités est active et MediaTailor ne remplit pas les pauses publicitaires avant ou après la valeur de suppression disponible dans la fenêtre de visualisation du manifeste. Lorsque le mode est réglé sur `AFTER_LIVE_EDGE`, la suppression des publicités est active. MediaTailor ne remplit pas les interruptions de publicité pendant ou après la période de suppression, qui est le Live Edge plus la valeur de suppression disponible plus la durée de la mémoire tampon.

Valeur de suppression disponible

La valeur de suppression disponible est une durée de décalage de bord en HH:MM:SS temps réel. MediaTailor ne remplira pas les pauses publicitaires à cette heure-ci dans la fenêtre rétrospective du manifeste.

Mode d'insertion

Le mode Insertion contrôle si les joueurs peuvent utiliser l'insertion d'annonces cousue ou guidée. La valeur par défaut `STITCHED_ONLY`, force toutes les sessions de jeu à utiliser l'insertion d'annonces Stitched (côté serveur). Paramètre `PLAYER_SELECT` permettant InsertionMode aux joueurs de sélectionner l'insertion d'annonces cousues ou guidées au moment de l'initialisation de la session. La valeur par défaut pour les joueurs qui ne spécifient pas de mode d'insertion est `Stitched`.

Réglages avancés

Les paramètres avancés facultatifs sont les suivants. Vous pouvez les configurer dans la MediaTailor console, à l'aide du AWS Command Line Interface (AWS CLI) ou à l'aide de l' API MediaTailor.

CDN content segment prefix (Préfixe du segment de contenu CDN)

Permet AWS Elemental MediaTailor de créer des manifestes contenant des URL vers votre chemin CDN pour les segments de contenu. Avant d'effectuer cette étape, configurez une règle dans votre réseau de diffusion de contenu (CDN) pour extraire les segments à partir de votre serveur d'origine. Pour Préfixe du segment du contenu CDN, saisissez le chemin du préfixe CDN.

Pour plus d'informations sur l'intégration MediaTailor à un CDN, consultez [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#).

CDN ad segment prefix (Préfixe du segment publicitaire CDN)

Permet AWS Elemental MediaTailor de créer des manifestes contenant des URL vers votre propre chemin CDN pour les segments publicitaires. Par défaut, MediaTailor diffuse des segments d'annonces provenant d'une CloudFront distribution Amazon interne avec des paramètres de cache par défaut. Avant de pouvoir compléter le champ CDN ad segment prefix (Préfixe du segment publicitaire du CDN), vous devez configurer une règle de votre CDN pour extraire les segments publicitaires de l'origine suivante, comme dans l'exemple suivant.

```
https://segments.mediatailor.<region>.amazonaws.com
```

Dans CDN ad segment prefix (Préfixe du segment publicitaire CDN), entrez le nom de votre préfixe CDN dans la configuration.

Pour plus d'informations sur l'intégration MediaTailor à un CDN, consultez [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#).

DASH origin manifest type (Type du manifeste d'origine DASH)

Si votre serveur d'origine produit des manifestes DASH à période unique, ouvrez la liste déroulante et choisissez SINGLE_PERIOD (PÉRIODE UNIQUE). Par défaut, MediaTailor gère les manifestes DASH comme des manifestes à plusieurs périodes. Pour de plus amples informations, veuillez consulter [the section called “Intégrer une MPEG-DASH source”](#).

DASH mpd location (Emplacement mdp DASH)

(Facultatif si nécessaire pour DASH) Emplacement de la description de la présentation multimédia (mpd). Choisissez DÉSACTIVÉ pour la situation suivante :

- Vous configurez des règles de routage CDN pour accéder aux MediaTailor manifestes.
- Vous utilisez les rapports côté client ou votre lecteur prend en charge les redirections HTTP persistantes.

Pour en savoir plus sur la fonction Location, consultez [the section called “Fonctionnalité de localisation”](#).

Nom du profil de transcodage

Nom qui associe cette configuration à un profil de transcodage personnalisé. Ce nom remplace les valeurs par défaut du transcodage dynamique de MediaTailor. Remplissez ce champ uniquement si vous avez déjà configuré des profils personnalisés à l'aide d'AWS Support.

Passthrough des marqueurs publicitaires

Pour HLS, active ou désactive la transmission des marqueurs publicitaires. Lorsque la transmission des marqueurs publicitaires est activée, les marqueurs publicitaires MediaTailor passent EXT-X-CUE-INEXT-X-CUE-OUT, et les marqueurs EXT-X-SPLICEPOINT-SCTE35 publicitaires passent du manifeste d'origine au manifeste MediaTailor personnalisé. Aucune logique n'est appliquée aux valeurs des marqueurs publicitaires ; elles sont transmises du manifeste d'origine au manifeste personnalisé telles quelles. Par exemple, si la valeur EXT-X-CUE-OUT est égale 60 à dans le manifeste d'origine, mais qu'aucune annonce n'est diffusée, la valeur MediaTailor sera pas modifiée pour qu'elle 0 apparaisse dans le manifeste personnalisé.

Délai d'expiration des requêtes ADS individuelles

Durée maximale, en millisecondes, d' MediaTailor attente d'une seule réponse HTTP lors de la résolution d'une décision publicitaire pendant la diffusion en direct ou en VOD. Valeur par défaut : 3000. Plage valide : 250 —6000.

Pour plus d'informations sur la gestion de MediaTailor l'assemblage des publicités dans les flux de travail en direct et en VOD, consultez [Comportement d'assemblage des publicités en direct](#) et [Comportement d'assemblage publicitaire pour la VOD](#)

Ce réglage correspond au `AdsRequestTimeoutMilliseconds` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationTimeouts` paramètre.

Nombre maximal de demandes ADS simultanées

Le nombre de requêtes simultanées que MediaTailor vous pouvez adresser à l'ADS. Valeur par défaut : 1. Plage valide : 1 —100.

Ce réglage correspond au `MaxConcurrentAdsRequests` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationConcurrency` paramètre.

Durée maximale de personnalisation des ADS en direct

Budget temporel total, en millisecondes, pour toutes les activités ADS lors de la personnalisation du manifeste en direct. Ce budget est partagé entre toutes les pauses publicitaires dans une seule demande manifeste. Valeur par défaut : 10000. Plage valide : 250 —10000.

Pour plus d'informations sur le comportement d'assemblage des publicités en direct, consultez [Comportement d'assemblage des publicités en direct](#).

Ce réglage correspond au `LiveMaximumAdsPersonalizationTimeMilliseconds` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationTimeouts` paramètre.

Durée maximale de personnalisation des ADS en VOD

Budget temporel total, en millisecondes, pour toutes les activités ADS lors de la personnalisation du manifeste VOD. Ce budget est partagé entre toutes les pauses publicitaires dans une seule demande manifeste. Valeur par défaut : 10000. Plage valide : 250 —10000.

Pour plus d'informations sur le comportement d'assemblage des publicités VOD, consultez [Comportement d'assemblage publicitaire pour la VOD](#)

Ce réglage correspond au `VodMaximumAdsPersonalizationTimeMilliseconds` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationTimeouts` paramètre.

Activer la parallélisation VOD VAST

Lorsque vous activez ce paramètre, MediaTailor traite les demandes ADS en parallèle pour les flux de travail VOD qui renvoient des réponses VAST. Ce paramètre s'applique uniquement aux flux de travail VOD. MediaTailor désactive ce paramètre par défaut. Pour prendre effet, définissez le `nombre maximum de requêtes ADS simultanées` sur une valeur supérieure à 1.

Plafonnement de fréquence avec demandes simultanées

Lorsque vous activez la parallélisation VAST, certains serveurs publicitaires peuvent ne pas appliquer de limites de fréquence pour les demandes d'annonces simultanées. Vérifiez auprès de votre fournisseur de serveur publicitaire si le plafonnement de fréquence est pris en charge dans des conditions de demande parallèle. Si votre serveur publicitaire ne prend pas en charge les limites de fréquence pour les demandes

simultanées, envisagez de passer à VMAP, qui gère correctement le plafonnement de fréquence pour les demandes d'annonces parallèles.

Ce réglage correspond au `EnableVodVastParallelization` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationConcurrency` paramètre.

Délai d'expiration de la requête ADS Prefetch

Durée maximale, en millisecondes, d' MediaTailor attente d'une seule réponse HTTP lors de la résolution d'une décision publicitaire lors de la récupération préalable. Par défaut : valeur du délai d'expiration de la requête ADS individuelle. Plage valide : 250 —10000.

Pour plus d'informations sur le préchargement, consultez. [Publicités de prélecture](#)

Ce réglage correspond au `PrefetchAdsRequestTimeoutMilliseconds` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationTimeouts` paramètre.

Temps de personnalisation ADS maximal par Prefetch

Budget temporel total, en millisecondes, pour toutes les activités ADS lors de la récupération préalable. Par défaut, ce délai n'a pas de limite supérieure. Plage valide : 250 —10000.

Pour plus d'informations sur le préchargement, consultez. [Publicités de prélecture](#)

Ce réglage correspond au `PrefetchMaximumAdsPersonalizationTimeMilliseconds` membre du [PutPlaybackConfiguration](#) `AdsPersonalizationTimeouts` paramètre.

Régler les délais d'attente ensemble

Pour chaque flux de travail, vous devez définir ensemble le délai d'attente individuel et le budget de temps total. Le délai d'attente individuel ne peut pas dépasser le budget total. Les flux de travail en direct et en VOD partagent un seul délai d'expiration des requêtes ADS individuelles. Définissez ce délai d'attente avec les budgets de temps totaux en direct et en VOD. De même, définissez simultanément le délai d'expiration de la requête Prefetch ADS et le délai de personnalisation ADS maximal de Prefetch. Vous ne pouvez pas définir l'un sans l'autre.

Affichage des détails MediaTailor de configuration

En plus des valeurs fournies lorsque la configuration a été créée, AWS Elemental MediaTailor affiche le nom de la configuration, les points de terminaison de la lecture et les URL d'accès pertinentes. Pour afficher une configuration, procédez comme suit.

Pour afficher une configuration

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Sur la page Configurations, choisissez le Nom de configuration de la configuration à afficher.

Modification des paramètres MediaTailor de configuration

Vous pouvez modifier une configuration pour mettre à jour le mappage du serveur d'origine et du serveur de décision publicitaire (ADS), ou modifier la façon dont vous AWS Elemental MediaTailor interagissez avec un réseau de distribution de contenu (CDN).

Pour modifier une configuration

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Sur la page Configurations, choisissez le nom de la configuration que vous souhaitez modifier.
3. Sur la page Détails de configuration, choisissez Edit (Modifier), puis revoyez les paramètres de configuration en fonction des besoins. Vous ne pouvez pas modifier le nom de la configuration. Pour plus d'informations sur les attributs de configuration, consultez [Création d'une configuration MediaTailor de lecture](#).
4. Choisissez Enregistrer.

Supprimer des MediaTailor configurations

Vous pouvez supprimer une configuration pour la rendre indisponible à la lecture AWS Elemental MediaTailor.

Pour supprimer une configuration

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.

2. Sur la page Configurations, procédez comme suit :
 - Choisissez le nom de la configuration que vous souhaitez supprimer.
 - Dans la colonne Configuration name (Nom de configuration), choisissez l'option située en regard du nom, puis choisissez Delete (Supprimer).
3. Dans la zone de confirmation Delete (Supprimer), entrez **Delete**, puis choisissez Delete (Supprimer).

Intégrer une source de contenu pour l'insertion d' MediaTailor annonces

Cette rubrique couvre l'intégration de différents types de sources de contenu vidéo avec MediaTailor. MediaTailor prend en charge les protocoles de diffusion HLS et DASH pour le contenu en direct et à la demande. Le service peut insérer ou remplacer des publicités pendant les pauses publicitaires désignées et a des exigences spécifiques en matière de structure et de formatage des manifestes vidéo d'entrée afin de permettre ces fonctionnalités. Les rubriques suivantes fournissent des informations détaillées sur les exigences relatives aux sources d'entrée et les étapes à suivre pour intégrer les contenus HLS et DASH afin MediaTailor de permettre des expériences publicitaires personnalisées.

Rubriques

- [Exigences relatives à la source d'entrée pour l'insertion d' MediaTailor annonces](#)
- [Intégrer une source HLS](#)
- [Intégrer une MPEG-DASH source](#)
- [Sécurisation AWS Elemental MediaTailor interactions d'origine avec SIGv4](#)

Exigences relatives à la source d'entrée pour l'insertion d' MediaTailor annonces

Une source d'entrée doit répondre aux exigences suivantes pour fonctionner MediaTailor :

- Utilisez Apple HLS (HTTP Live Streaming) ou MPEG DASH (Dynamic Adaptive Streaming over HTTP)
- Utiliser le streaming en direct ou la vidéo à la demande (VOD)
- Être accessible sur l'Internet public et avoir une adresse IP publique

- Utilisez des ports HTTP standard (port 80) ou des ports HTTPS (port 443). MediaTailor ne prend pas en charge les ports personnalisés pour les communications avec le serveur d'origine.
- Contient des marqueurs publicitaires dans l'un des formats décrits dans le didacticiel de [démarrage avec l'insertion d' annonces MediaTailor](#)

Intégrer une source HLS

AWS Elemental MediaTailor prend en charge .m3u8 les manifestes HLS avec un 3 ou plus pour EXT-X-VERSION la diffusion en direct et la vidéo à la demande (VOD). En MediaTailor cas de coupure de publicité, il tente d'insérer ou de remplacer une annonce, en fonction du type de contenu. S'il n'y a pas assez d'annonces pour couvrir la durée, le flux de contenu sous-jacent ou la liste configurée MediaTailor s'affiche pendant le reste de la pause publicitaire. Pour plus d'informations sur le comportement des publicités HLS en fonction du type de contenu, consultez [Comprendre le comportement d'insertion des AWS Elemental MediaTailor publicités](#).

Les sections suivantes fournissent plus d'informations sur la façon dont les MediaTailor manifestes HLS sont gérés.

Rubriques

- [Marqueurs publicitaires compatibles HLS](#)
- [Activation de la transmission des marqueurs publicitaires](#)
- [Gestion des balises de manifeste HLS](#)
- [Exemples de manifestes HLS](#)

Marqueurs publicitaires compatibles HLS

AWS Elemental MediaTailor identifie les limites de disponibilité des publicités dans un manifeste HLS en analysant le manifeste d'entrée pour détecter les marqueurs publicitaires pris en charge. Les sections suivantes décrivent les marqueurs utilisés par MediaTailor .

EXT-X-ASSET

La balise EXT-X-ASSET contient des métadonnées utilisées par le serveur ADS (ad decision server) afin de personnaliser le contenu pour le spectateur. Les paramètres EXT-X-ASSET sont des paires clé-valeur séparées par des virgules.

Pour utiliser cette balise, vous devez remplir les conditions suivantes :

- Vous devez URL-encoder utiliser les EXT-X-ASSET valeurs du manifeste d'origine. L'exemple suivant montre la EXT-X-ASSET balise avec des clés et URL-encodés des valeurs.

```
#EXT-X-ASSET:GENRE=CV,CAID=12345678,EPISODE="Episode%20Name%20Date",SEASON="Season%20Name%20and%20Number",SERIES="Series%2520Name"
```

- Vous devez inclure la [asset.] variable dynamique et les clés dans votre configuration MediaTailor ADS. L'exemple suivant montre une configuration MediaTailor ADS utilisant la [asset.] variable dynamique et les touches.

```
https://myads.com/stub?c=[asset.GENRE]&g=[asset.CAID]&e=[asset.EPISODE]&s=[asset.SEASON]&k=[asset.SERIES]
```

Exemple de demande VAST

L'exemple suivant présente une demande VAST GET à un serveur ADS.

```
https://myads.com/stub?c=CV&g=12345678&e=Episode%20Name%20Date&s=Season%20Name%20and%20Number&k=Series%2520Name
```

EXT-X-CUE-OUT et EXT-X-CUE-IN

Ce type de marqueur de publicité est le plus courant. Les exemples suivants illustrent des options pour ces marqueurs de file d'attente.

```
#EXT-X-CUE-OUT:DURATION=120
```

```
...
```

```
#EXT-X-CUE-IN
```

```
#EXT-X-CUE-OUT:30.000
```

```
...
```

```
#EXT-X-CUE-IN
```

```
#EXT-X-CUE-OUT
```

...

#EXT-X-CUE-IN

EXT-X-DATERANGE

Avec les balises de marqueur publicitaire EXT-X-DATERANGE, vous utilisez les attributs SCTE35-OUT pour spécifier la durée de la diffusion de publicités.

Note

AWS Elemental MediaTailor ignore tous START-DATE les attributs fournis pour les marqueurs EXT-X-DATERANGE publicitaires.

Vous pouvez spécifier la diffusion de publicités de l'une des manières suivantes :

- Balise EXT-X-DATERANGE avec spécifications SCTE35-OUT et DURATION.

Exemple

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",DURATION=60.000,SCTE35-OUT=0xF
```

- Les balises EXT-X-DATERANGE associées, la première avec une spécification SCTE35-OUT et la seconde avec une spécification SCTE35-IN.

Exemple

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",SCTE35-IN=0xF
```

- Une combinaison des options précédentes. Vous spécifiez une balise EXT-X-DATERANGE avec les spécifications SCTE35-OUT et DURATION suivies d'une balise EXT-X-DATERANGE avec une spécification SCTE35-IN. Dans ce cas, MediaTailor utilise le premier paramètre de repérage parmi les deux spécifications.

Exemple

```
#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2019-01T00:15:00Z
\",DURATION=60.000,SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2019-01T00:15:00Z\",SCTE35-
IN=0xF
```

EXT-X-SPLICEPOINT-SCTE35

Vous ajoutez la balise de marqueur EXT-X-SPLICEPOINT-SCTE35 publicitaire avec une SCTE-35 charge utile en binaire codé en base64. Le binaire décodé doit SCTE-35 `splice_info_section` contenir le marqueur de repérage `0x34`, pour le début de l'opportunité de placement du fournisseur, et le marqueur d'entrée, pour la fin de l'opportunité de placement du fournisseur. `0x35`

L'exemple suivant illustre la spécification de point de jointure avec des charges utiles binaires encodées en base64 qui spécifient des marqueurs sortants de file d'attente et entrants en file d'attente.

```
#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAxMTMyMjE5M190NAAmXz5JA==
...
#EXT-X-SPLICEPOINT-SCTE35:/DA4AAAAAAAAAAP/wBQb+tTaaAwAiAiBDVUVJAAAKqH+/
DBFNRFNOQjAwMTEzMjIxOTJfTjUAAIiGK1s=
```

Activation de la transmission des marqueurs publicitaires

Par défaut, pour HLS, les manifestes MediaTailor personnalisés n'incluent pas les marqueurs SCTE-35 publicitaires des manifestes d'origine. Lorsque le transfert des marqueurs publicitaires est activé, MediaTailor passe par les marqueurs publicitaires suivants depuis les manifestes d'origine vers les manifestes personnalisés :

- EXT-X-CUE-IN
- EXT-X-CUE-OUT
- EXT-X-SPLICEPOINT-SCTE35

La transmission des marqueurs publicitaires est un paramètre facultatif. Utilisez le relais des marqueurs publicitaires si vous souhaitez que les marqueurs publicitaires SCTE soient inclus dans le manifeste MediaTailor personnalisé. Les cas d'utilisation courants sont les suivants :

- Remplacement du contenu : effectuez le remplacement ou la restriction du contenu.
- Suivi des publicités : déclenchez des informations de suivi des publicités en fonction de la présence ou de l'absence d'un ou de plusieurs marqueurs publicitaires.
- Paramètres du lecteur : activez la fonctionnalité de nettoyage ou de compte à rebours dans l'interface utilisateur du lecteur, en fonction de la présence ou de l'absence de marqueurs publicitaires.

 Note

MediaTailor ne modifie pas les valeurs de ces marqueurs. Par exemple, si la valeur EXT-X-CUE-OUT est égale 60 à dans le manifeste d'origine, mais qu'aucune annonce n'est diffusée, la valeur ne MediaTailor sera pas modifiée pour qu'elle 0 apparaisse dans le manifeste personnalisé.

Activer le transfert des marqueurs publicitaires

Vous pouvez activer le transfert des marqueurs publicitaires à l'aide du Console de gestion AWS ou du AWS Command Line Interface (AWS CLI).

Pour activer le transfert des marqueurs publicitaires à l'aide de la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Sélectionnez Nouvelle configuration ou Modifier la configuration.
3. Dans la section Paramètres avancés, sélectionnez Activer dans le menu déroulant.

Pour activer le transfert des marqueurs publicitaires à l'aide du AWS Command Line Interface (AWS CLI)

Utilisez la commande [put-playback-configuration](#).

Gestion des balises de manifeste HLS

Cette section décrit comment AWS Elemental MediaTailor gérer les balises dans le manifeste de sortie personnalisé.

EXT-X-CUE balises

MediaTailor remplace EXT-X-CUE-OUTEXT-X-CUE-OUT-CONT, et EXT-X-CUE-IN balises dans le manifeste d'entrée par des EXT-X-DISCONTINUITY balises dans le manifeste de sortie. Les balises DISCONTINUITY marquent les limites suivantes :

- Où le contenu principal passe à une publicité
- Où une publicité passe à un autre publicité
- Où une publicité repasse au contenu principal

EXT-X-DATERANGE tags

MediaTailor passe par les EXT-X-DATERANGE balises du manifeste d'entrée au manifeste de sortie. MediaTailor insère également des EXT-X-DISCONTINUITY balises qui correspondent aux DATERANGE balises. Les balises DISCONTINUITY marquent les limites suivantes :

- Où le contenu principal passe à une publicité
- Où une publicité passe à un autre publicité
- Où une publicité repasse au contenu principal

EXT-X-KEY tags

MediaTailor passe par les EXT-X-KEY balises du manifeste d'entrée. Ces balises indiquent que le contenu principal est chiffré. Étant donné que les publicités ne sont pas chiffrées, MediaTailor insère EXT-X-KEY:METHOD=NONE au début d'une diffusion de publicités. Lorsque la lecture revient au contenu principal, le chiffrement est MediaTailor réactivé en insérant la EXT-X-KEY balise dont la METHOD valeur est définie comme type de cryptage.

Tags non reconnus

MediaTailor passe par toutes les balises inconnues et personnalisées du manifeste d'entrée au manifeste de sortie.

Exemples de manifestes HLS

Les sections suivantes fournissent des exemples de manifestes d'origine et de manifestes personnalisés HLS. La compréhension de ces exemples peut vous aider à configurer et à résoudre les problèmes liés à vos MediaTailor flux de travail.

Pour plus d'informations sur la façon dont les paramètres de requête sont appliqués aux manifestes et aux segments HLS, consultez [MediaTailor Initialisation de session HLS implicite](#).

Comprendre les types de playlist HLS

Le HTTP Live Streaming (HLS) utilise deux principaux types de playlists :

Liste de lecture multivariante

Une liste de lecture multivariante est le fichier d'index de premier niveau qui répertorie tous les rendus disponibles du contenu. Il contient des références à des listes de lecture multimédia, mais ne contient aucun segment multimédia en lui-même. Cette playlist permet aux joueurs de sélectionner le rendu le plus approprié en fonction des conditions du réseau, des capacités de l'appareil ou des préférences de l'utilisateur.

Ce type de playlist est également connu sous plusieurs autres noms dans divers contextes, notamment la playlist principale, le manifeste principal, la liste de lecture principale, la liste de lecture principale, le fichier d'index ou le master M3U8.

Dans MediaTailor les flux de travail, la playlist multivariante est le point d'entrée pour les demandes de lecture et c'est là que commence la personnalisation des publicités.

Liste de lecture multimédia

Une liste de lecture multimédia contient les informations relatives au segment multimédia correspondant à un rendu spécifique (niveau de qualité) du contenu. Il inclut les informations de chronométrage URLs, les segments et les autres métadonnées nécessaires à la lecture d'un rendu unique.

Ce type de playlist est également connu sous le nom de playlist multimédia, child manifest, chunklist, media M3U8 ou playlist de rendu.

Dans MediaTailor les flux de travail, les playlists multimédias sont personnalisées pour inclure à la fois des segments de contenu et des segments publicitaires dans le bon ordre.

Pour plus d'informations sur les types de playlist HLS, consultez [Types de listes de lecture HLS](#).

Exemples de manifestes d'origine HLS

L'exemple suivant montre une liste de lecture multivariante HLS AWS Elemental MediaTailor reçue par HLS depuis l'origine du contenu.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-STREAM-INF:BANDWIDTH=2665726,AVERAGE-
BANDWIDTH=2526299,RESOLUTION=960x540,FRAME-
RATE=29.970,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=3956044,AVERAGE-
BANDWIDTH=3736264,RESOLUTION=1280x720,FRAME-
RATE=29.970,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=995315,AVERAGE-
BANDWIDTH=951107,RESOLUTION=640x360,FRAME-
RATE=29.970,CODECS="avc1.4D401E,mp4a.40.2",SUBTITLES="subtitles"
index_3.m3u8
#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-
ID="subtitles",NAME="caption_1",DEFAULT=YES,AUTOSELECT=YES,FORCED=NO,LANGUAGE="eng",URI="index_
```

Dans cet exemple de playlist multivariante :

- Les `#EXT-X-STREAM-INF` balises définissent différents rendus avec des résolutions et des débits binaires variables
- Chaque rendu fait référence à une playlist multimédia (telle que `index_1.m3u8`)
- La `#EXT-X-MEDIA` balise définit une piste de sous-titre

L'exemple suivant montre une liste de lecture multimédia HLS AWS Elemental MediaTailor reçue par HLS depuis l'origine du contenu. Cet exemple utilise les balises `EXT-X-CUE-OUT` et `EXT-X-CUE-IN` pour décrire les opportunités de diffusion de publicités.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:8779957
#EXTINF:6.006,
```



```

index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
index_1_8779958.ts?m=1566416212
#EXTINF:5.372,
index_1_8779959.ts?m=1566416212
#EXT-0ATCLS-SCTE35:/DA1AAAAAsvhAP/wFAXwAAAGf+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXT-X-CUE-OUT:20.020
#EXTINF:0.634,
index_1_8779960.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=0.634,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:6.006,
index_1_8779961.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=6.640,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:6.006,
index_1_8779962.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=12.646,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:6.006,
index_1_8779963.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=18.652,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:1.368,
index_1_8779964.ts?m=1566416212
#EXT-X-CUE-IN
#EXTINF:4.638,
index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
index_1_8779968.ts?m=1566416212

```

Dans cet exemple de playlist multimédia :

- Les #EXTINF balises indiquent la durée de chaque segment
- Le #EXT-X-CUE-OUT tag marque le début d'une pause publicitaire
- Les #EXT-X-CUE-OUT-CONT balises fournissent des informations sur la pause publicitaire en cours
- Le #EXT-X-CUE-IN tag marque la fin du break publicitaire

Exemples de manifestes personnalisés HLS

L'exemple suivant montre une liste de lecture multivariante HLS personnalisée. AWS Elemental MediaTailor

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI="../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/ee1696a8-4f7f-4c4c-99de-9821131847e8/3.m3u8",GROUP-ID="subtitles",DEFAULT=YES,NAME="caption_1"
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-BANDWIDTH=2526299,RESOLUTION=960x540,SUBTITLES="subtitles",FRAME-RATE=29.97,BANDWIDTH=2665726
../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/ee1696a8-4f7f-4c4c-99de-9821131847e8/0.m3u8
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-BANDWIDTH=3736264,RESOLUTION=1280x720,SUBTITLES="subtitles",FRAME-RATE=29.97,BANDWIDTH=3956044
../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/ee1696a8-4f7f-4c4c-99de-9821131847e8/1.m3u8
#EXT-X-STREAM-INF:CODECS="avc1.4D401E,mp4a.40.2",AVERAGE-BANDWIDTH=951107,RESOLUTION=640x360,SUBTITLES="subtitles",FRAME-RATE=29.97,BANDWIDTH=995315
../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/ee1696a8-4f7f-4c4c-99de-9821131847e8/2.m3u8
```

Remarquez comment MediaTailor a modifié la liste de lecture multimédia URLs pour inclure des informations spécifiques à la session qui permettent l'insertion d'annonces personnalisées.

L'exemple suivant montre une playlist multimédia AWS Elemental MediaTailor personnalisée.

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:8779957
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/e309fffd02ba8498d864dcaacff7a5ad9/index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
```

```

https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779958.ts?m=1566416212
#EXTINF:5.372,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779959.ts?m=1566416212
#EXT-X-DISCONTINUITY
#EXTINF:3.066667,
../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779960
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779961
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779962
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779963
#EXTINF:2.966667,
../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779964
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779963.ts?m=1566416212
#EXTINF:1.368,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779964.ts?m=1566416212
#EXTINF:4.638,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779968.ts?m=1566416212

```

Dans cet exemple de playlist multimédia personnalisée :

- MediaTailor a inséré des segments publicitaires entre les segments de contenu
- Les #EXT-X-DISCONTINUITY balises marquent les transitions entre le contenu et les publicités
- Les segments de contenu sont diffusés depuis le serveur d'origine (MediaPackage dans cet exemple)
- Les segments publicitaires sont diffusés à partir de MediaTailor l'espace de stockage des segments publicitaires

Principales différences entre les manifestes personnalisés

Lors de la MediaTailor personnalisation des manifestes HLS, plusieurs modifications importantes sont apportées :

Modifications de listes de lecture multivariantes

- Les variantes de playlist URLs sont réécrites pour pointer vers MediaTailor -managed URLs qui incluent des informations de session
- L'ordre des balises peut être réorganisé pour une lecture optimale

Variantes de modifications de playlist

- Les marqueurs publicitaires (EXT-X-CUE-OUT,EXT-X-CUE-IN) sont remplacés par de véritables segments publicitaires
- Des marqueurs de discontinuité (EXT-X-DISCONTINUITY) sont ajoutés aux limites content/ad
- URLs Les segments de contenu sont réécrits pour pointer vers l'origine ou le CDN
- URLs Les segments publicitaires sont ajoutés au stockage des segments publicitaires MediaTailor du point vers

La compréhension de ces modifications peut vous aider à résoudre les problèmes liés à vos MediaTailor flux de travail et à garantir une configuration correcte de votre CDN et de votre lecteur.

Rubriques en relation

Pour plus d'informations sur les manifestes HLS MediaTailor, consultez les rubriques suivantes :

- [Types de listes de lecture HLS](#)- Explication détaillée des types de playlist HLS
- [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#)- Informations sur l'utilisation d'un CDN avec MediaTailor

- [Comment fonctionne MediaTailor l'insertion d'annonces](#)- Vue d'ensemble du fonctionnement de l'insertion d' MediaTailor annonces

Intégrer une MPEG-DASH source

AWS Elemental MediaTailor prend en charge les manifestes .mpd en direct et en vidéo à la demande (VOD) qui suivent les directives du profil dynamique DASH. MediaTailor accepte les entrées de manifeste à périodes multiples et à période unique, et fournit des sorties de DASH-compliant manifeste à périodes multiples DASH-compliant.

Les manifestes d'entrée doivent comporter les éléments suivants :

- SCTE-35 flux d'événements avec paramètres d'informations d'épissage pour l'un `splice insert` ou `time signal` l'autre. Les paramètres peuvent être fournis en XML clair ou en binaire encodé en base64.
- Segment templates avec segment timelines.

Pour les manifestes publiés, MediaTailor nécessite que les mises à jour depuis le serveur d'origine laissent les éléments suivants inchangés :

- Heures de début de période, spécifiées dans l'attribut `start`.
- Valeurs de `presentationTimeOffset` dans les modèles de segment des représentations de la période.

Il est recommandé de donner à l'annonce la même disponibilité `AdaptationSet` et les mêmes `Representation` paramètres que les périodes de diffusion du contenu. AWS Elemental MediaTailor utilise ces paramètres pour transcoder les publicités afin qu'elles correspondent au flux de contenu, afin de passer facilement de l'un à l'autre.

Les sections suivantes fournissent plus d'informations sur la façon dont les publicités MediaTailor sont gérées dans les manifestes DASH.

Rubriques

- [Marqueurs publicitaires DASH](#)
- [DASH et durée de disponibilité](#)
- [SCTE-35 marqueurs et limites de période dans DASH](#)

- [Numérotation des segments du manifeste DASH](#)
- [Exemples de MPD DASH](#)
- [Fonction de localisation DASH](#)

Marqueurs publicitaires DASH

MediaTailor

AWS Elemental MediaTailor utilise SCTE-35 des marqueurs de repère pour identifier les annonces disponibles dans le manifeste DASH selon la logique suivante :

- Multi-period DASH : MediaTailor insère des publicités pour la première Event dans chacune d'entre elles Period contenant l'un SpliceInsert ou l'EventTimeSignal autre des marqueurs. MediaTailor ignore les Event marqueurs supplémentaires dans le Period
- Single-period DASH : MediaTailor insère des publicités Period contenant Event l'un ou l'autre des marqueurs SpliceInsert ou des marqueurs de TimeSignal découpage.

Par défaut, AWS Elemental MediaTailor gère les manifestes DASH comme des manifestes à plusieurs périodes. Vous pouvez modifier votre configuration afin de gérer des manifestes DASH à période unique à partir de votre serveur d'origine. Pour plus d'informations, consultez [the section called "Création d'une configuration"](#).

Les sections suivantes fournissent des informations supplémentaires sur la gestion des marqueurs publicitaires DASH et fournissent des manifestes décorés depuis l'origine.

Exigences XML relatives au manifeste d'origine DASH

Les marqueurs publicitaires figurant dans les manifestes DASH depuis l'origine doivent être correctement formatés pour pouvoir MediaTailor identifier les pauses publicitaires. Les rubriques suivantes décrivent ces exigences de mise en forme en XML clair.

SpliceInsert en XML clair

SpliceInsert les marqueurs publicitaires en XML clair doivent contenir les éléments suivants :

- EventStream doit avoir l'attribut suivant : `schemeIdUri=urn:scte:scte35:2013:xml`
- Event doit tenir `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` doit tenir `scte35:SpliceInsert`

- `scte35:SpliceInsert` doit avoir l'attribut suivant : `outOfNetworkIndicator="true"`

Exemple **SpliceInsert** en XML

Dans l'exemple suivant, les marqueurs SCTE requis sont en gras.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832" tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="1350000"/>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
    .
    .
    .
  </Period>
```

TimeSignal en XML clair

`TimeSignal` les marqueurs publicitaires en XML clair doivent contenir les éléments suivants :

- `EventStream` doit avoir l'attribut suivant : `schemeIdUri=urn:scte:scte35:2013:xml`
- `Event` doit tenir `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` doit tenir `scte35:TimeSignal`
- `scte35:SpliceInfoSection` doit également contenir `scte35:SegmentationDescriptor`
- `scte35:SegmentationDescriptor` doit avoir l'attribut suivant, dont la valeur est valide [Cue-out chiffres](#) : `segmentationTypeId="xx"`
- `scte35:SegmentationDescriptor` doit tenir `scte35:SegmentationUpid`

Cue-out chiffres

Les numéros de repère suivants sont pris en charge pour le `segmentationTypeId`

Message de segmentation	Valeur segmentationTypeId	Valeur hexadécimale
Fin de la publicité pour les distributeurs	51	0x51
Début de la publicité pour les distributeurs	50	0x32
Fin de l'opportunité de placement pour les distributeurs	55	0x37
Début de l'opportunité de placement pour les distributeurs	54	0x36
Fin de pause	35	0x23
Fin de la publicité du fournisseur	49	0x31
Début de la publicité du fournisseur	48	0x30
Fin de l'opportunité de placement par superposition de fournisseurs	57	0x39
Début de l'opportunité de placement par superposition de fournisseurs	56	0x38
Fin de l'opportunité de placement des prestataires	53	0x35
Début de l'opportunité de placement de prestataires	52	0x34
Pause de départ	34	0x22

Exemple **TimeSignal** en XML

Dans l'exemple suivant, les marqueurs SCTE requis sont en gras.

```
<Period start="PT346530.250S" id="178443" duration="PT61.561S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="5310000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183003" tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="3442857000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="1414668"
segmentationEventCancelIndicator="false"
segmentationDuration="8100000" segmentationTypeId="52" segmentNum="0"
segmentsExpected="0">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="false"
noRegionalBlackoutFlag="false" archiveAllowedFlag="false" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="12"
segmentationUpidLength="2">0100</scte35:SegmentationUpid>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>
    .
    .
    .
  </EventStream>
</Period>
```

Exigences binaires codées en base64 du manifeste d'origine DASH

Les marqueurs publicitaires figurant dans les manifestes DASH depuis l'origine doivent être correctement formatés pour pouvoir MediaTailor identifier les pauses publicitaires. Les rubriques suivantes décrivent ces exigences de formatage en binaire codé en base64.

Les marqueurs `SpliceInsert` publicitaires `TimeSignal` contenus dans les manifestes codés en base64 et les marqueurs publicitaires doivent contenir les éléments suivants :

- `EventStream` doit avoir l'attribut suivant : `urn:scte:scte35:2014:xml+bin`
- `Event` doit tenir `scte35:Signal`
- `scte35:Signal` doit contenir un `scte35:Binary` fichier binaire codé en base64.

Le binaire décodé doit `splice_info_section` fournir les mêmes informations que celles requises pour des marqueurs publicitaires XML clairs.

- Le type de commande doit être `splice_insert()` soit `time_signal()`
- Les paramètres supplémentaires doivent être conformes à ceux décrits dans [TimeSignal en XML clair](#) et [SpliceInsert en XML clair](#).

Le binaire décodé doit fournir un `splice_info_section` avec le même ensemble d'informations que le XML clair fournirait dans un élément `scte35:SpliceInfoSection`. La commande doit être de type `splice_insert()` ou `time_signal()` et les paramètres supplémentaires doivent être conformes à ceux décrits précédemment pour une livraison en XML clair.

L'exemple suivant illustre cette option, avec les marqueurs requis en gras.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
    <Event presentationTime="1541436240" duration="24" id="29">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>/DAhAAAAAAAAAAP/wEAUAAAHaf+9/fgAg9YDAAAAAAAAA25aoh</
scte35:Binary>
      </scte35:Signal>
    </Event>
    <Event presentationTime="1541436360" duration="24" id="30">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">

      <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgaW5jb2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYmluYXJ5Lg=
scte35:Binary>
      </scte35:Signal>
    </Event>
    .
    .
    .
  </Period>
```

Voici le binaire décodé pour le premier événement répertorié dans l'exemple précédent. Le paramètre pour `splice_command_type` est 5, ce qui indique `splice_insert`.

```
{
  "table_id": 252,
  "section_syntax_indicator": false,
  "private_indicator": false,
```

```

    "section_length": 33,
    "protocol_version": 0,
    "encrypted_packet": false,
    "encryption_algorithm": 0,
    "pts_adjustment": 0,
    "cw_index": 0,
    "tier": "0xFFF",
    "splice_command_length": 16,
    "splice_command_type": 5,
    "splice_command": {
        "splice_event_id": 448,
        "splice_event_cancel_indicator": false,
        "out_of_network_indicator": true,
        "program_splice_flag": true,
        "duration_flag": true,
        "splice_immediate_flag": false,
        "utc_splice_time": {
            "time_specified_flag": false,
            "pts_time": null
        },
    },
    "component_count": 0,
    "components": null,
    "break_duration": {
        "auto_return": false,
        "duration": {
            "pts_time": 2160000,
            "wall_clock_seconds": 24.0,
            "wall_clock_time": "00:00:24:000000"
        }
    },
    },
    "unique_program_id": 49152,
    "avail_num": 0,
    "avails_expected": 0
},
"splice_descriptor_loop_length": 0,
"splice_descriptors": null,
"Scte35Exception": {
    "parse_status": "SCTE-35 cue parsing completed with 0 errors.",
    "error_messages": [],
    "table_id": 252,
    "splice_command_type": 5
}
}

```

DASH et durée de disponibilité

Pendant la lecture, lorsqu' AWS Elemental MediaTailor une annonce apparaît, elle remplace une partie ou la totalité de l'offre par des publicités. MediaTailor lance le remplacement des annonces au début de la disponibilité de l'annonce et inclut les annonces suivantes :

- Si l'annonce indique une durée, MediaTailor incluez autant d'annonces que possible dans la limite de durée, sans remplacer le contenu qui suit.
- Si aucune durée n'est fournie, MediaTailor inclut les annonces jusqu'à la fin de la période de disponibilité de l'annonce. Pour des manifestes à plusieurs périodes, il s'agit de la fin de la période. Pour les manifestes à période unique, c'est la fin de l'événement. MediaTailor ne diffuse pas les publicités au-delà de la fin de la période de disponibilité de l'annonce et, lorsqu'elle atteint la fin, tronque l'annonce en cours au lieu de remplacer le contenu qui suit.

Comment ? AWS Elemental MediaTailor recherche la durée de disponibilité de l'annonce

AWS Elemental MediaTailor recherche un paramètre de durée dans l'ordre suivant :

1. Event duration
2. Pour l'insertion de jointure, le paramètre `scte35:BreakDuration duration`
3. Pour le signal temporel, le paramètre `scte35:SegmentationDescriptor segmentationDuration`

S'il AWS Elemental MediaTailor ne trouve aucun de ces paramètres, il gère l'inclusion des annonces sans définir de durée.

L'exemple suivant illustre un Event qui a une duration.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></
scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="1350000"/>

```

```

    </scte35:SpliceInsert>
  </scte35:SpliceInfoSection>
</Event>
...

```

L'exemple suivant illustre une diffusion de publicités sans durée spécifiée. L' Event n'a pas de duration et l'élément `scte35:SpliceInsert` ne contient pas d'élément enfant `scte35:BreakDuration`.

```

<Period start="PT444836.720S" id="123597" duration="PT12.280S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event>
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531856"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5675385600"/></
scte35:Program>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
    ...
  </EventStream>
</Period>

```

SCTE-35 marqueurs et limites de période dans DASH

MediaTailor prend en charge les origines DASH qui fournissent des manifestes à période unique ou à périodes multiples. Pour les manifestes à période unique, MediaTailor convertit le manifeste en un format à périodes multiples en utilisant les informations temporelles provenant du `EventStream` pour définir les limites des périodes. Après la conversion, les manifestes à période unique et à périodes multiples sont soumis au même flux de personnalisation des publicités.

Lorsqu'il MediaTailor détecte un événement de pause publicitaire linéaire au cours d'une période, il traite l'ensemble de la période comme une interruption publicitaire et la personnalise en conséquence. MediaTailor utilise l'heure de début de la période (`Period@start`) comme heure de début de pause publicitaire, quel que soit le décalage temporel signalé lors de l'événement lui-même.

Comportements clés

MediaTailor applique les règles suivantes lors du traitement des SCTE-35 événements pendant les périodes DASH :

- Pour les pauses publicitaires linéaires, si le calendrier d'un événement indique que la pause publicitaire commence après la limite de la période, MediaTailor ignore l'heure de début signalée et commence la personnalisation à `Period@start`. Pour les événements non linéaires (superposition), MediaTailor respecte le signal. `presentationTime`
- Si un nouvel événement SCTE apparaît au milieu d'un saut publicitaire linéaire existant, il est MediaTailor ignoré.
- Si un événement non linéaire (superposition) actif existe déjà dans la période d'origine, MediaTailor ignore tout nouvel événement linéaire de cette période.

Signalisation d'un événement en milieu de période

Avec la AWS Elemental MediaPackage v2, MediaTailor reçoit SCTE-35 des marqueurs même lorsqu'aucune nouvelle période n'est créée à la limite du marqueur. Cela peut entraîner l'apparition de marqueurs SCTE au milieu des périodes de contenu.

Lorsqu'un nouvel SCTE-35 événement signale une interruption publicitaire linéaire au milieu d'une période de contenu, il MediaTailor reconnaît qu'il s'agit d'une opportunité de pause publicitaire, mais lance la personnalisation au moment de l'événement `Period@start` plutôt qu'au moment indiqué. Cette incompatibilité peut entraîner les problèmes suivants :

- Des publicités qui n'apparaissent jamais au spectateur
- Ardoise insérée pour combler une période restante créée par le décalage horaire

Calcul du décalage de l'événement depuis le début de la période

Utilisez la formule suivante pour déterminer dans quelle mesure un événement est décalé par rapport au début de sa période :

$$\text{Offset} = (\text{presentationTime} - \text{presentationTimeOffset}) / \text{timescale}$$

L'exemple suivant montre un `EventStream presentationTime` dont la valeur est décalée par rapport au début de la période :

```
<Period id="1741948755359" start="PT10522H39M15.359S">
  <EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="900000"
    presentationTimeOffset="156775387982400">
    <Event presentationTime="156775484008800" duration="369000000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
```

```

    <scte35:TimeSignal>
      <scte35:SpliceTime ptsTime="587770208"/>
    </scte35:TimeSignal>
    <scte35:SegmentationDescriptor segmentationEventId="168602131"
      segmentationEventCancelIndicator="false"
      segmentationDuration="36900000"
      segmentationTypeId="34"
      segmentNum="0" segmentsExpected="1">
    </scte35:SegmentationDescriptor>
  </scte35:SpliceInfoSection>
</Event>
</EventStream>
</Period>

```

Pour cet exemple, le calcul du décalage est le suivant :

$$156775484008800 - 156775387982400 = 96026400 \text{ ticks}$$

$$96026400 / 90000 = 1066.96 \text{ seconds (approximately 17 minutes 46 seconds)}$$

La durée de pause annoncée est de :

$$36900000 / 90000 = 410 \text{ seconds (approximately 6 minutes 50 seconds)}$$

Comme l'événement ne commence pas à la limite de la période, la personnalisation MediaTailor commence à `Period@start`, soit plus de 17 minutes avant la pause prévue. Cela se traduit par des publicités qui ne correspondent pas à la fenêtre d'interruption réelle. Si l'ardoise est configurée, MediaTailor insère des périodes d'ardoise pour combler le vide.

Repérage précoce

Un signal anticipé se produit lorsqu'une pause publicitaire prend fin avant la durée initialement indiquée. Dans un manifeste multipériode, cela crée une nouvelle période qui met fin prématurément à la pause publicitaire initiale.

Le comportement varie d'une AWS Elemental MediaPackage version à l'autre :

- MediaPackage v1 — Crée la nouvelle période et supprime toutes les informations relatives aux événements qu'elle contient.
- MediaPackage v2 — Inclut deux marqueurs publicitaires dans le EventStream : le découpage d'origine et le repérage. Pour signaler le signal, réglez `suroutOfNetworkIndicator` à `false`.

MediaTailor reconnaît le premier signal en vérifiant la `outOfNetworkIndicator` valeur. Lorsqu'il MediaTailor trouve un indice qui correspond à un indice enregistré au cours de la même période, il ne traite pas le signal comme une nouvelle opportunité de rupture publicitaire.

L'exemple suivant montre un EventStream avec à la fois un signal de sortie et un signal d'entrée précoce :

```
<EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="90000"
  presentationTimeOffset="276012000">
  <Event presentationTime="273308400" duration="5400000">
    <scte35:SpliceInfoSection duration="5400000" protocolVersion="0"
      ptsAdjustment="183600" tier="4095">
      <scte35:SpliceInsert spliceEventId="222" spliceEventCancelIndicator="false"
        outOfNetworkIndicator="true" spliceImmediateFlag="false"
        uniqueProgramId="1" availNum="1" availsExpected="1">
        <scte35:Program>
          <scte35:SpliceTime ptsTime="273124800"/>
        </scte35:Program>
        <scte35:BreakDuration autoReturn="true" duration="5400000"/>
      </scte35:SpliceInsert>
    </scte35:SpliceInfoSection>
  </Event>
  <Event presentationTime="276012000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183600" tier="4095">
      <scte35:SpliceInsert spliceEventId="1" spliceEventCancelIndicator="false"
        outOfNetworkIndicator="false" spliceImmediateFlag="false"
        uniqueProgramId="1" availNum="1" availsExpected="1">
        <scte35:Program>
          <scte35:SpliceTime ptsTime="275828400"/>
        </scte35:Program>
      </scte35:SpliceInsert>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
```

Le premier événement (`outOfNetworkIndicator="true"`) signale le début de la pause publicitaire. Le deuxième événement (`outOfNetworkIndicator="false"`) signale le début de la pause qui met fin à la pause.

Atténuer les problèmes liés aux marqueurs à mi-période SCTE-35

Si vous constatez un comportement de personnalisation publicitaire inattendu en raison de l'apparition de SCTE-35 marqueurs en milieu de période dans vos diffusions en direct DASH, MediaTailor vous pouvez supprimer ou supprimer des marqueurs spécifiques qui ne sont pas applicables à votre flux de travail de personnalisation des annonces.

Lorsque la suppression des marqueurs est activée, MediaTailor effectue les actions suivantes :

- **Supprimer les marqueurs** : MediaTailor ignore les marqueurs spécifiés et supprime les événements correspondants du manifeste généré. S'il ne reste aucun autre événement, supprime MediaTailor également le vide EventStream de la période.
- **Marqueurs de passage** : MediaTailor ignore les marqueurs spécifiés à des fins de personnalisation, mais les marqueurs restent dans le manifeste généré.

Les types de coupures suivants sont pris en charge pour la suppression ou le transfert :

- BREAK
- PROVIDER_ADVERTISEMENT
- DISTRIBUTOR_ADVERTISEMENT
- PROVIDER_PLACEMENT_OPPORTUNITY
- DISTRIBUTOR_PLACEMENT_OPPORTUNITY
- PROVIDER_OVERLAY_PLACEMENT_OPPORTUNITY

MediaTailor enregistre chaque marqueur supprimé ou transmis et émet une métrique de comptage lorsqu'un ou plusieurs marqueurs sont supprimés dans une réponse manifeste.

Important

Cette configuration s'applique uniquement aux manifestes DASH en mode diffusion en direct. Pour configurer la suppression des marqueurs ou le passthrough en fonction de votre configuration, contactez le AWS support. Lorsque vous contactez l'assistance, spécifiez les types de pauses que vous souhaitez supprimer ou supprimer.

Numérotation des segments du manifeste DASH

MediaTailor prend en charge les segments multimédia définis à <SegmentTemplate> l'aide de l'attribut <SegmentTimeline> et. Vous pouvez spécifier la liste des segments multimédias dans l'attribut media à l'aide de l'identifiant \$Number\$ ou \$Time\$.

L'exemple suivant illustre un SegmentTemplate avec un paramètre d'attribut media qui utilise l'identifiant \$Number\$.

```
<SegmentTemplate initialization="index_subtitles_4_0_init.mp4?m=1532451703" media="index_subtitles_4_0_$Number$.mp4?m=1532451703"
  presentationTimeOffset="1062336677920" startNumber="2349899" timescale="90000">
  <SegmentTimeline>
    <S d="540540" r="2" t="1062338840080"/>
    <S d="69069" t="1062340461700"/>
  </SegmentTimeline>
</SegmentTemplate>
```

L'exemple suivant illustre un SegmentTemplate avec un paramètre d'attribut media qui utilise l'identifiant \$Time\$.

```
<SegmentTemplate
  initialization="asset_720p_8000K_9_init.mp4" media="asset_720p_8000K_9_$Time$.mp4"
  startNumber="1" timescale="90000">
  <SegmentTimeline>
    <S d="180000" r="2" t="0"/>
    <S d="147000" t="540000"/>
  </SegmentTimeline>
</SegmentTemplate>
```

Exemples de MPD DASH

Les sections suivantes fournissent des exemples d'origine MPDs et de personnalisation de DASH MPDs. La compréhension de ces exemples peut vous aider à configurer et à résoudre les problèmes liés à vos MediaTailor flux de travail.

Pour plus d'informations sur la façon dont les paramètres de requête sont appliqués aux manifestes et aux segments DASH, consultez [MediaTailor Initialisation implicite de session DASH](#).

Comprendre la structure du DASH MPD

Le streaming adaptatif dynamique via HTTP (DASH) utilise un manifeste MPD (Media Presentation Description) pour diffuser du contenu en streaming. Le MPD est un document XML qui décrit la structure et la disponibilité du contenu multimédia.

MPD (description de la présentation aux médias)

Le MPD est le principal fichier manifeste du streaming DASH qui décrit la structure et la disponibilité du contenu multimédia. Il contient des informations sur les périodes, les ensembles d'adaptation, les représentations et les segments qui constituent le contenu diffusé en continu.

Ce type de manifeste est également connu sous plusieurs autres noms dans divers contextes, notamment le manifeste DASH, le DASH MPD, le manifeste principal (par rapport au HLS) ou le manifeste de présentation.

Dans MediaTailor les flux de travail, le MPD est le point d'entrée pour les demandes de diffusion et c'est là que commence la personnalisation des publicités.

Period

Une période est une section temporelle d'une présentation DASH. Chaque période contient un ou plusieurs ensembles d'adaptation et représente une période médiatique. Dans les flux de travail d'insertion d'annonces, des périodes distinctes sont généralement utilisées pour faire la distinction entre le contenu et les publicités.

Dans MediaTailor les flux de travail, les périodes sont utilisées pour séparer le contenu principal du contenu publicitaire, chaque annonce étant généralement représentée par sa propre période.

AdaptationSet

Un AdaptationSet regroupe un ensemble de versions codées interchangeables d'un ou de plusieurs composants de contenu multimédia. Par exemple, l'un AdaptationSet peut contenir plusieurs niveaux de qualité vidéo, tandis qu'un autre peut contenir plusieurs options de langue audio.

Dans les MediaTailor flux de travail, AdaptationSets ils sont préservés lors de l'insertion des publicités afin de garantir la cohérence des types de médias entre le contenu et les publicités.

Représentation

Une représentation est une version codée spécifique du contenu multimédia dans un AdaptationSet. Chaque représentation diffère généralement en termes de débit, de résolution

ou d'autres paramètres de codage, ce qui permet aux clients de sélectionner la version la plus appropriée en fonction des conditions du réseau et des capacités de l'appareil.

Dans MediaTailor les flux de travail, les représentations dans les périodes publicitaires sont mises en correspondance aussi étroitement que possible avec les représentations dans les périodes de contenu afin de garantir une expérience de visionnage fluide.

Pour des informations plus détaillées sur les types de manifestes DASH, consultez [Types de manifestes DASH](#).

Pour plus d'informations sur la structure et la configuration du manifeste DASH dans AWS Elemental MediaPackage, consultez la section du guide de MediaPackage l'utilisateur sur la présentation de DASH.

Exemples de MPD DASH en direct

Cette section fournit des exemples de DASH en direct MPDs. Chaque exemple répertorie un MPD tel qu'il a été reçu du serveur d'origine et MediaTailor a ensuite personnalisé le MPD avec des publicités.

Exemple d'insert d'épissure DASH MPD

Exemple de MPD DASH Origin pour un insert Splice

L'exemple suivant, tiré d'un MPD, montre une publicité publiée dans un manifeste reçu par DASH depuis l'origine du contenu. Cet exemple utilise le `SpliceInsert` marqueur pour indiquer la disponibilité d'une annonce.

```
<Period start="PT173402.036S" id="46041">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="9450000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183265" tier="4095">
        <scte35:SpliceInsert spliceEventId="99" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="7835775000"/></scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="9450000"/>
        </scte35:SpliceInsert>
        <scte35:SegmentationDescriptor segmentationEventId="99"
segmentationEventCancelIndicator="false" segmentationDuration="9450000">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="true"
noRegionalBlackoutFlag="true" archiveAllowedFlag="true" deviceRestrictions="3"/>
        </scte35:SegmentationDescriptor>
      </Event>
    </EventStream>
  </Period>
```

```

        <scte35:SegmentationUpid segmentationUpidType="8" segmentationUpidLength="0"/>
    >
        <scte35:SegmentationTypeID segmentationType="52"/>
        <scte35:SegmentNum segmentNum="1"/>
        <scte35:SegmentsExpected segmentsExpected="1"/>
    </scte35:SegmentationDescriptor>
</scte35:SpliceInfoSection>
</Event>
</EventStream>
<AdaptationSet mimeType="video/mp4" segmentAlignment="true"
subsegmentAlignment="true" startWithSAP="1" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1000000" codecs="avc1.4D401F">
        <SegmentTemplate timescale="30000" media="index_video_1_0_$Number
$.mp4?m=1528475245" initialization="index_video_1_0_init.mp4?m=1528475245"
startNumber="178444" presentationTimeOffset="10395907501">
            <SegmentTimeline>
                <S t="10395907501" d="60060" r="29"/>
                <S t="10397709301" d="45045"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Representation id="2" bandwidth="96964" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="48000" media="index_audio_2_0_$Number
$.mp4?m=1528475245" initialization="index_audio_2_0_init.mp4?m=1528475245"
startNumber="178444" presentationTimeOffset="16633452001">
            <SegmentTimeline>
                <S t="16633452289" d="96256" r="3"/>
                <S t="16633837313" d="95232"/>
                <S t="16633932545" d="96256" r="4"/>
                <S t="16634413825" d="95232"/>
                <S t="16634509057" d="96256" r="5"/>
                <S t="16635086593" d="95232"/>
                <S t="16635181825" d="96256" r="4"/>
                <S t="16635663105" d="95232"/>
                <S t="16635758337" d="96256" r="5"/>
                <S t="16636335873" d="71680"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>

```

```
</AdaptationSet>
</Period>
```

Dans cet exemple de MPD d'origine :

- L'<EventStream>élément contient des marqueurs SCTE-35 qui indiquent la disponibilité des publicités
- L'<scte35:SpliceInsert>élément fournit des détails sur la disponibilité de la publicité
- L'<scte35:BreakDuration>élément indique la durée de la pause publicitaire
- Les <AdaptationSet> éléments définissent les flux vidéo et audio disponibles

Exemple de MPD personnalisé DASH pour insert Splice

AWS Elemental MediaTailor personnalise les produits publicitaires avec des spécifications publicitaires. Les personnalisations reflètent les données de l'utilisateur qui sont reçues du lecteur et les campagnes de publicité actuellement en cours.

L'exemple suivant montre une annonce disponible après l'avoir AWS Elemental MediaTailor personnalisée.

```
<Period id="178443_1" start="PT96H15M30.25S">
  <BaseURL>http://111122223333.cloudfront.net/nbc_fallback_2/</BaseURL>
  <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="10000000" codecs="avc1.640028" height="1080" id="1"
width="1920">
      <SegmentTemplate initialization="nbc_fallback_ad_2_1080p_10init.mp4"
media="nbc_fallback_ad_2_1080p_10_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180000" r="13" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="4000000" codecs="avc1.64001f" height="720" id="2"
width="1280">
      <SegmentTemplate initialization="nbc_fallback_ad_2_720p_9init.mp4"
media="nbc_fallback_ad_2_720p_9_${Number%09d$.mp4" startNumber="1" timescale="90000">
```

```

    <SegmentTimeline>
      <S d="180000" r="13" t="0"/>
      <S d="176940" t="2520000"/>
    </SegmentTimeline>
  </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
  <Representation id="8" bandwidth="128000" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <SegmentTemplate initialization="nbc_fallback_ad_2_audio_2init.mp4"
media="nbc_fallback_ad_2_audio_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
      <SegmentTimeline>
        <S d="180000" r="13" t="0"/>
        <S d="176940" t="2520000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
</Period>

```

Dans cet exemple de MPD personnalisé :

- MediaTailor a créé une nouvelle période pour le contenu de l'annonce
- L'<BaseURL>élément pointe vers l'emplacement du contenu publicitaire
- Les <AdaptationSet> éléments conservent une structure similaire à celle du contenu
- Les <Representation> éléments fournissent différents niveaux de qualité pour le contenu publicitaire

Exemple de signal horaire DASH MPD

Exemple de MPD d'origine DASH pour le signal horaire

L'exemple suivant, tiré d'un MPD, montre une publicité publiée dans un manifeste reçu par DASH depuis l'origine du contenu. Cet exemple utilise le `TimeSignal` marqueur pour indiquer la disponibilité d'une annonce.

```

<Period start="PT173402.036S" id="46041">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">

```

```

    <Event duration="9450000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183265" tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="7835775000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="99"
segmentationEventCancelIndicator="false" segmentationDuration="9450000">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="true"
noRegionalBlackoutFlag="true" archiveAllowedFlag="true" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="8" segmentationUpidLength="0"/>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
  <AdaptationSet mimeType="video/mp4" segmentAlignment="true"
subsegmentAlignment="true" startWithSAP="1" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1000000" codecs="avc1.4D401F">
      <SegmentTemplate timescale="30000" media="index_video_1_0_${Number}
$.mp4?m=1528475245" initialization="index_video_1_0_init.mp4?m=1528475245"
startNumber="178444" presentationTimeOffset="10395907501">
        <SegmentTimeline>
          <S t="10395907501" d="60060" r="29"/>
          <S t="10397709301" d="45045"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>

```

Dans cet exemple de MPD d'origine :

- L'<scte35:TimeSignal>élément est utilisé à la place de <scte35:SpliceInsert>
- <scte35:SegmentationDescriptor>fournit des informations supplémentaires sur la disponibilité de la publicité

Exemple de MPD personnalisé DASH pour le signal horaire

AWS Elemental MediaTailor personnalise les produits publicitaires avec des spécifications publicitaires. Les personnalisations reflètent les données de l'utilisateur qui sont reçues du lecteur et les campagnes de publicité actuellement en cours.

L'exemple suivant montre une annonce disponible après l'avoir AWS Elemental MediaTailor personnalisée.

```
<Period id="178443_1" start="PT96H15M30.25S">
  <BaseURL>http://111122223333.cloudfront.net/nbc_fallback_2/</BaseURL>
  <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="10000000" codecs="avc1.640028" height="1080" id="1"
width="1920">
      <SegmentTemplate initialization="nbc_fallback_ad_2_1080p_10init.mp4"
media="nbc_fallback_ad_2_1080p_10_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180000" r="13" t="0"/>
          <S d="176940" t="252000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
```

Le MPD personnalisé pour le signal horaire est similaire à celui pour l'insert Splice, avec la MediaTailor création d'une nouvelle période pour le contenu publicitaire.

Exemples de MPD VOD DASH

Cette section fournit des exemples de vidéo à la demande (VOD) DASH MPDs. Chaque exemple répertorie un MPD tel qu'il a été reçu du serveur d'origine et MediaTailor a ensuite personnalisé le MPD avec des publicités.

Les VOD DASH MPDs suivent la même structure que les vidéos en direct MPDs, mais elles ont généralement un `type="static"` attribut dans l'élément MPD et peuvent contenir plusieurs périodes pour différents segments de contenu.

Pour des exemples de VOD DASH MPDs, consultez la MediaTailor documentation sur [Marqueurs publicitaires DASH](#).

Principales différences en matière de personnalisation MPDs

Lors de la MediaTailor personnalisation de DASH MPDs, il apporte plusieurs modifications importantes :

Gestion des périodes

- De nouvelles périodes sont créées pour le contenu publicitaire
- Les heures de début des périodes sont ajustées pour maintenir la continuité de la chronologie
- EventStream les éléments dotés de marqueurs SCTE-35 sont traités et retirés

AdaptationSet et gestion des représentations

- AdaptationSets dans l'annonce, des périodes sont créées pour correspondre au contenu AdaptationSets
- Des représentations sont créées pour différents niveaux de qualité du contenu publicitaire
- SegmentTemplate les éléments sont mis à jour pour pointer vers le contenu de l'annonce

La compréhension de ces modifications peut vous aider à résoudre les problèmes liés à vos MediaTailor flux de travail et à garantir une configuration correcte de votre CDN et de votre lecteur.

Rubriques en relation

Pour plus d'informations sur DASH MPDs MediaTailor, consultez les rubriques suivantes :

- [Types de manifestes DASH](#)- Explication détaillée des types de manifestes DASH
- [Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu](#)- Informations sur l'utilisation d'un CDN avec MediaTailor
- [Comment fonctionne MediaTailor l'insertion d'annonces](#)- Vue d'ensemble du fonctionnement de l'insertion d' MediaTailor annonces
- Pour des informations complètes sur la structure et la MediaPackage configuration du manifeste DASH, consultez la section du guide de MediaPackage l'utilisateur sur la présentation de DASH.

Fonction de localisation DASH

Cette section fournit des informations sur la fonctionnalité d'emplacement pour DASH, qui est activée par défaut dans AWS Elemental MediaTailor. Lisez cette section si vous créez des règles de routage du réseau de diffusion de contenu (CDN) pour accéder aux MediaTailor manifestes. Lisez aussi cette

section si vous utilisez le reporting côté serveur avec des lecteurs qui ne prennent pas en charge les redirections HTTP permanentes.

Qu'est-ce que la fonctionnalité d'emplacement ?

La fonctionnalité d'emplacement permet aux lecteurs qui ne prennent pas en charge les redirections HTTP permanentes de fournir un comportement permanent dans leurs demandes de mise à jour du manifeste.

AWS Elemental MediaTailor utilise une initialisation sans session et nécessite un comportement de redirection HTTP persistant de la part de ses joueurs. Avec les rapports côté serveur, lorsque le joueur fait une demande de mise à jour du manifeste MediaTailor, le service émet une redirection temporaire 302, pour diriger le joueur vers un point de terminaison pour le manifeste personnalisé. MediaTailor inclut un identifiant de session dans la réponse, en tant que paramètre de requête. L'intention est pour le lecteur de suivre l'URL pendant l'intégralité de la session, mais les lecteurs ne prenant pas en charge les redirections HTTP permanentes abandonnent la redirection et reviennent à l'URL d'origine. Lorsqu'un joueur revient à l'URL d'origine, une nouvelle session est créée pour chaque nouvelle demande au lieu de conserver la session d'origine. Cela peut entraîner une corruption du manifeste.

La spécification DASH fournit une solution à ce problème dans la fonction de localisation, qui est activée par défaut dans les AWS Elemental MediaTailor configurations. Lorsque cette fonctionnalité est activée, MediaTailor place l'URL absolue dans la <Location> balise manifeste. Les lecteurs qui ne prennent pas en charge les redirections HTTP permanentes peuvent utiliser l'URL fournie dans <Location> pour demander les mises à jour au manifeste.

Est-ce que j'ai besoin de désactiver la fonctionnalité d'emplacement de ma configuration ?

La fonctionnalité d'emplacement remplace les règles de routage CDN que vous configurez pour accéder aux manifestes de AWS Elemental MediaTailor . Par conséquent, vous pouvez avoir besoin de la désactiver. La fonctionnalité d'emplacement n'a pas d'incidence sur la mise en cache CDN du contenu ou des segments de publicité.

Recherchez votre situation dans la liste suivante pour déterminer si vous avez besoin de désactiver la fonctionnalité d'emplacement dans votre configuration et comment la traiter :

- Si vous n'avez pas de règles d'acheminement de CDN pour accéder aux manifestes de AWS Elemental MediaTailor , laissez le paramètre d'emplacement activé.
- Sinon, utilisez les règles suivantes :

- Si vous n'utilisez pas le reporting côté serveur ou que vos lecteurs prennent tous en charge les redirections HTTP permanentes, désactivez la fonctionnalité d'emplacement. Pour plus d'informations sur la façon de procéder sur la console, consultez [the section called “Création d'une configuration”](#).
- Sinon, contactez [AWS Support](#).

Dois-je utiliser la fonctionnalité d'emplacement ?

Vous devez utiliser la fonction de localisation pour les joueurs qui ne prennent pas en charge les redirections HTTP persistantes. Utilisez l'URL fournie dans la balise <Location> pour toutes les demandes de mise à jour de votre manifeste.

Exemple

Exemples d'URL et exemple de balise <Location>.

- Exemple Exemple d'URL de demande initiale

```
https://777788889999.mediataylor.us-east-1.amazonaws.com/v1/
dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd
```

- Exemple Exemple de réponse 302 redirigée

```
/v1/dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd?
aws.sessionId=0e5d9b45-ae97-49eb-901b-893d043e0aa6
```

- Exemple Exemple de balise d'emplacement dans un manifeste

```
<Location>https://777788889999.mediataylor.us-east-1.amazonaws.com/v1/
dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd?
aws.sessionId=0e5d9b45-ae97-49eb-901b-893d043e0aa6</Location>
```

Sécurisation AWS Elemental MediaTailor interactions d'origine avec SIGv4

Signature Version 4 (Sigv4) est un protocole de signature utilisé pour authentifier les MediaTailor demandes vers les origines prises en charge via HTTPS. MediaTailor prend uniquement en charge la communication HTTPS et n'autorise pas les connexions HTTP. Avec la signature SIGv4, MediaTailor inclut un en-tête d'autorisation signé dans la demande d'origine HTTPS adressée à MediaTailor Channel Assembly, Amazon S3 et AWS Elemental MediaPackage la version 2.

Vous pouvez utiliser SIGv4 à votre origine pour vous assurer que les demandes de manifeste ne sont traitées que si elles proviennent MediaTailor et contiennent un en-tête d'autorisation signé. De cette façon, les configurations MediaTailor de lecture non autorisées sont empêchées d'accéder à votre contenu d'origine. Si l'en-tête d'autorisation signé est valide, votre origine répond à la demande. Si elle n'est pas valide, la demande échoue.

Les sections suivantes décrivent les exigences relatives à l'utilisation de la signature MediaTailor SIGv4 pour les origines prises en charge.

MediaTailor Exigences relatives à l'assemblage des canaux

Si vous utilisez SIGv4 pour protéger l'origine de votre MediaTailor Channel Assembly, les conditions suivantes doivent être remplies MediaTailor pour accéder au manifeste :

- L'URL de base d'origine de votre MediaTailor configuration doit être un canal Channel Assembly au format suivant : `channel-assembly.mediatailor.region.amazonaws.com`
- Votre origine doit être configurée pour utiliser le protocole HTTPS. MediaTailor prend uniquement en charge la communication HTTPS et n'autorise pas les connexions HTTP. Si le protocole HTTPS n'est pas activé à l'origine, la demande ne MediaTailor sera pas signée.
- Votre chaîne doit disposer d'une politique d'accès à l'origine qui inclut les éléments suivants :
 - Accès principal pour MediaTailor accéder à votre chaîne. Accordez l'accès à `mediatailor.amazonaws.com`.
 - Autorisations IAM `mediatailor` : `GetManifest` pour lire toutes les playlists multivariantes référencées par la configuration. MediaTailor

Pour plus d'informations sur la définition d'une politique sur la chaîne, consultez [Création d'une chaîne à l'aide de la MediaTailor console](#).

Exemple politique d'accès à l'origine pour Channel Assembly, limitée au compte de MediaTailor configuration

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediatailor:GetManifest",
  "Resource": "arn:aws:mediatailor:us-west-2:777788889999:channel/ca-origin-channel",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "777788889999"}
```

```
}  
}
```

Exemple politique d'accès à l'origine pour Channel Assembly, limitée à la configuration de MediaTailor lecture

```
{  
  "Effect": "Allow",  
  "Principal": {"Service": "mediatailor.amazonaws.com"},  
  "Action": "mediatailor:GetManifest",  
  "Resource": "arn:aws:mediatailor:us-west-2:777788889999:channel/ca-origin-channel",  
  "Condition": {  
    "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-west-2:777788889999:playbackConfiguration/test"}  
  }  
}
```

Exigences relatives à Amazon S3

Si vous utilisez Sigv4 pour protéger votre origine Amazon S3, les conditions suivantes doivent être remplies MediaTailor pour accéder au manifeste :

- L'URL de base d'origine MediaTailor de votre configuration doit être un compartiment S3 au format suivant : `s3.region.amazonaws.com`
- Votre origine doit être configurée pour utiliser le protocole HTTPS. MediaTailor prend uniquement en charge la communication HTTPS et n'autorise pas les connexions HTTP. Si le protocole HTTPS n'est pas activé à l'origine, la demande ne MediaTailor sera pas signée.
- Votre chaîne doit disposer d'une politique d'accès à l'origine qui inclut les éléments suivants :
 - Accès principal pour MediaTailor accéder à votre compartiment. Accordez l'accès à `mediatailor.amazonaws.com`.

Pour plus d'informations sur la configuration de l'accès dans IAM, consultez la section Gestion des [accès](#) dans le guide de l'utilisateur d'AWS Identity and Access Management.

- Autorisations IAM `s3 : GetObject` pour lire tous les manifestes de niveau supérieur référencés par la MediaTailor configuration.

Pour des informations générales sur SIGv4 pour Amazon S3, consultez la rubrique [Demandes d'authentification \(AWS Signature version 4\) dans la](#) référence de l'API Amazon S3.

Exemple politique d'accès à l'origine pour Amazon S3, limitée au compte MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::mybucket/*",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "111122223333"}
  }
}
```

Exemple politique d'accès à l'origine pour Amazon S3, limitée à la configuration de MediaTailor lecture

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::mybucket/*",
  "Condition": {
    "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-west-2:111122223333:playbackConfiguration/test"}
  }
}
```

MediaPackage exigences

Si vous utilisez SIGv4 pour protéger votre origine MediaPackage v2, les conditions suivantes doivent être remplies MediaTailor pour accéder au manifeste :

- L'URL de base d'origine dans votre MediaTailor configuration doit être un point de terminaison MediaPackage v2 au format suivant : `mediapackagev2.region.amazonaws.com`
- Votre origine doit être configurée pour utiliser le protocole HTTPS. MediaTailor prend uniquement en charge la communication HTTPS et n'autorise pas les connexions HTTP. Si le protocole HTTPS n'est pas activé à l'origine, la demande ne MediaTailor sera pas signée.
- Votre chaîne doit disposer d'une politique d'accès à l'origine qui inclut les éléments suivants :
 - Accès principal pour MediaTailor accéder à votre terminal. Accordez l'accès à `mediatailor.amazonaws.com`.

- Autorisations IAM `mediapackagev2` : `GetObject` pour lire toutes les playlists multivariantes référencées par la configuration. MediaTailor

Pour des informations générales sur SIGv4 pour MediaPackage v2, consultez la rubrique [Demandes d'authentification \(AWS Signature version 4\) dans la](#) référence de l'API MediaPackage v2.

Exemple politique d'accès à l'origine pour la MediaPackage v2, limitée au compte MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediapackagev2:GetObject",
  "Resource": "arn:aws:mediapackagev2:us-west-2:444455556666:channelGroup/emp-origin-channel-group/channel/emp-origin-channel/originEndpoint/emp-origin-endpoint",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "444455556666"}
  }
}
```

Exemple politique d'accès à l'origine pour la MediaPackage v2, limitée à la configuration de MediaTailor lecture

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediapackagev2:GetObject",
  "Resource": "arn:aws:mediapackagev2:us-west-2:444455556666:channelGroup/emp-origin-channel-group/channel/emp-origin-channel/originEndpoint/emp-origin-endpoint",
  "Condition": {
    "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-west-2:444455556666:playbackConfiguration/test"}
  }
}
```

Intégration AWS Elemental MediaTailor avec les plateformes publicitaires de Google

MediaTailor authentifie automatiquement les connexions de serveur à serveur avec Google Ad Manager (GAM), Google Campaign Manager (GCM) et Google Display & Video 360 (DV360).

Lorsqu'il MediaTailor détecte qu'une demande d'annonce ou un événement de suivi est destiné à un serveur publicitaire de Google, celui-ci établit automatiquement la connexion sécurisée et authentifiée requise.

Vous devez disposer d'un compte sur la plateforme publicitaire Google concernée. MediaTailor prend en charge les types d'intégration suivants :

- **Server-side intégration** : sécurise MediaTailor automatiquement les demandes d'annonces côté serveur et les demandes de suivi des impressions sur les plateformes publicitaires de Google. Cela inclut notamment les éléments suivants :
 - **Google Ad Manager (GAM)** : les demandes d'annonces adressées au serveur publicitaire de Google pour les éditeurs sont automatiquement sécurisées, ce qui est nécessaire pour accéder aux acheteurs autorisés, à savoir le site de vente d'annonces en temps réel et l'échange d'annonces de Google.
 - **Google Campaign Manager (GCM)** : Ad-serving et les demandes de suivi sont automatiquement acheminées via le point de terminaison authentifié de Google et sécurisées, ce qui permet aux annonceurs qui mènent des campagnes sur ces plateformes d'obtenir des rapports plus précis et de réduire le nombre d'impressions rejetées.
 - **Display & Video 360 (DV360)** : les demandes d'annonces sont automatiquement acheminées via le point de terminaison authentifié de Google et sécurisées, ce qui permet aux annonceurs qui mènent des campagnes sur ces plateformes d'obtenir des rapports plus précis et de réduire le nombre d'impressions rejetées.
- **Client-side intégration** : intégration d'un lecteur à l'aide du SDK PAL (Google Programmatic Access Libraries). Cette intégration est requise si vous souhaitez utiliser le type de transaction d'enchère ouverte dans Google Ad Manager.

Toutes les autres demandes d'annonces continuent de fonctionner sans modification.

La prise en charge par Ad Manager des types de transactions programmatiques varie en fonction du type d'intégration que vous utilisez. Pour obtenir la liste des options disponibles, consultez la section Types de [transactions](#) ou contactez l'équipe chargée de votre compte Google.

Rubriques

- [Server-side AWS Elemental MediaTailor intégration avec les plateformes publicitaires de Google](#)
- [Client-side AWS Elemental MediaTailor intégration avec Google Ad Manager](#)
- [Optimisation du taux de remplissage des annonces avec Google Ad Manager](#)

Server-side AWS Elemental MediaTailor intégration avec les plateformes publicitaires de Google

MediaTailor authentifie automatiquement les connexions de serveur à serveur avec Google Ad Manager (GAM), Google Campaign Manager (GCM) et Google Display & Video 360 (DV360) à l'aide de certificats SSL émis par Google. Lorsque MediaTailor détecte des demandes d'annonces ou des événements de suivi destinés aux serveurs publicitaires de Google, il établit automatiquement la connexion sécurisée et authentifiée requise. Aucun certificat SSL ou ticket d'assistance supplémentaire n'est requis.

Prérequis : vous devez disposer d'un compte actif sur la plateforme publicitaire Google concernée (Google Ad Manager, Google Campaign Manager ou Display & Video 360).

MediaTailor applique l'authentification par certificat Google-issued SSL en fonction des règles d'URI de destination suivantes :

Google Ad Manager (GAM) : demandes d'annonces côté serveur

- Cette règle s'applique aux demandes d'annonces adressées à `serverside.doubleclick.net`, notamment lorsque vous configurez l'URL du modèle ADS avec ce domaine ou lorsqu'une demande de redirection d'annonce est résolue vers ce domaine.
- MediaTailor ajoute le paramètre de `ssss=mediatailor` requête s'il n'est pas présent, puis sécurise la connexion à l'aide d'un certificat Google-issued SSL.
- Cela permet d'accéder aux acheteurs autorisés, à savoir le marché de vente d'annonces en temps réel et l'échange d'annonces de Google.

Google Ad Manager (GAM) : redirection des demandes d'annonces dans VMAP

- Cette règle se déclenche sur les demandes d'annonces `pubads.g.doubleclick.net` redirigées lorsque vous avez MediaTailor déjà vu des redirections VAST vers `serverside.doubleclick.net` dans la demande en cours et qu'une réponse ADS précédente était une réponse VMAP.
- MediaTailor change automatiquement le domaine `serverside.doubleclick.net` et établit une connexion authentifiée à l'aide du certificat Google-issued SSL.
- Cela permet d'accéder aux acheteurs autorisés pour les flux de travail VMAP-based publicitaires.

Google Campaign Manager (GCM) : diffusion d'annonces et demandes de suivi

- Cette règle se déclenche lors des demandes `ad.doubleclick.net` de diffusion d'annonces et de suivi lorsque l'URI contient `/ddm/pfadx/` ou `/ddm/trackimp/` (codé ou non codé).
- MediaTailor change automatiquement le domaine `serverside.doubleclick.net` et établit une connexion authentifiée à l'aide du certificat Google-issued SSL.
- Cela permet aux annonceurs de mener des campagnes sur ces plateformes en fournissant des rapports plus précis et en réduisant le nombre d'impressions rejetées.

Display & Video 360 (DV360) : demandes d'annonces

- Cette règle déclenche des demandes `bid.g.doubleclick.netactivation` ou `d'pagead2.googlesyndication.comannonce` lorsque l'URI contient `/dbm/vast/` (codé ou non codé).
- MediaTailor change automatiquement le domaine `serverside.doubleclick.net` et établit une connexion authentifiée à l'aide du certificat Google-issued SSL.
- Cela permet la diffusion d'annonces authentifiées pour les campagnes Display & Video 360.

Toutes les autres demandes d'annonces sont traitées sans authentification Google-issued par certificat SSL.

Note

L'authentification par certificat SSL est désormais automatique. Si vous avez déjà soumis un ticket d'assistance pour activer les certificats SSL, aucune action n'est nécessaire : vos configurations continuent de fonctionner.

Client-side AWS Elemental MediaTailor intégration avec Google Ad Manager

Une intégration MediaTailor côté client est requise pour utiliser les SDK PAL (Programmatic Access Libraries) de Google Ad Manager et pour utiliser le type de transaction d'enchère ouverte de Google Ad Manager.

Les SDK PAL fournissent des informations sur le contenu, l'appareil et les données utilisateur d'une session de lecture. Grâce au SDK PAL, vous pouvez fournir ces informations à Google Ad Manager, qui peut alors mieux déterminer les publicités ciblées à diffuser. Les SDK sont disponibles pour

Android, iOS, HTML5 et Cast. Pour plus d'informations sur l'utilisation des kits SDK PAL, consultez le SDK [PAL de Google Ad Manager](#).

Pour créer une intégration côté client avec Ad Manager

1. Utilisez le SDK PAL pour générer un nonce. Pour plus d'informations sur la configuration d'un nonce, consultez la documentation de votre SDK [sur Google Ad Manager PAL SDK](#).

 Note

Le nonce est une chaîne cryptée que PAL génère pour les demandes de flux. Chaque demande doit avoir un nonce unique.

2. Utilisez le givn paramètre de votre requête ADS pour transmettre la valeur nonce. Pour ce faire, mettez à jour l'URL de votre ADS afin d'y inclure `&givn=[player_params.givn]`. Pour obtenir des instructions, veuillez consulter [Activer le suivi côté client](#).

SDK pour lecteurs Datazoom

MediaTailor a établi un partenariat avec Datazoom pour fournir des SDK gratuits aux joueurs afin de faciliter l'intégration avec des SDK tels que ceux proposés dans le Google Ad Manager PAL. Pour plus d'informations sur le Datazoom et le MediaTailor partenariat, consultez [SDK pour lecteurs gratuits Datazoom](#). Pour accéder aux SDK du lecteur Datazoom, utilisez les coordonnées figurant sur le site Datazoom with [AWS](#)

Optimisation du taux de remplissage des annonces avec Google Ad Manager

Lorsque vous utilisez Google Ad Manager (GAM) comme serveur publicitaire MediaTailor, les paramètres d'URL de votre modèle ADS ont une incidence directe sur le taux de remplissage des annonces et les revenus. Les valeurs de paramètres incorrectes sont l'une des causes les plus fréquentes des faibles taux de remplissage.

Paramètres GAM recommandés

Incluez les paramètres suivants dans l'URL de votre modèle d'ADS pour garantir que GAM renvoie les annonces de manière cohérente et que les pauses publicitaires atteignent leur durée maximale.

`correlator=[avail.random]`

Garantit que GAM traite chaque pause publicitaire comme une demande unique et renvoie de nouvelles publicités. Sans ce paramètre, GAM peut renvoyer des réponses vides pour des pauses publicitaires répétées au cours d'une même session.

`pmxd=[session.avail_duration_ms]`

Durée maximale d'un pod publicitaire en millisecondes. Utilisez la variable MediaTailor dynamique `[session.avail_duration_ms]` pour que cette valeur corresponde automatiquement à la durée réelle du marqueur publicitaire SCTE. Ne codez pas cette valeur en dur : si la durée codée en dur est plus courte que la durée réelle de la pause publicitaire, le temps restant n'est pas utilisé.

`pmad=-1`

Le nombre maximum d'annonces par pod. Réglez-le -1 pour autoriser autant d'annonces que possible pour combler la pause, ou fixez-le à la limite souhaitée (par exemple 10). Une valeur faible (telle que 4) limite artificiellement le nombre d'annonces pouvant combler une pause, même lorsque davantage de stocks sont disponibles.

`vpos`

Position de la vidéo (pre-roll, midroll ou postroll). Définissez cette valeur sur la valeur qui correspond à votre type d'espace publicitaire réel. Ne pas utiliser `[avail.type]` : cette variable MediaTailor ne prend pas en charge cette variable et elle se transformera en une valeur vide, ce qui peut amener GAM à rejeter la demande ou à renvoyer un inventaire de faible valeur.

Paramètres du lecteur pour GAM

Certains paramètres GAM nécessitent des valeurs fournies par l'éditeur ou l'application du lecteur lors de l'initialisation de la session. Transmettez-les en MediaTailor tant que paramètres de lecteur et référencez-les dans l'URL de votre modèle ADS. Pour plus de détails sur la configuration des paramètres du lecteur, consultez [MediaTailor variables de lecteur pour les requêtes ADS](#).

`ppid=[player_params.ppid]`

L' Publisher-Provided identifiant programmatique. Fournissez cette valeur attribuée par l'éditeur afin que GAM puisse suivre les sessions et continuer à renvoyer des annonces à chaque pause publicitaire. Dans le cas contraire, GAM pourrait cesser de renvoyer des publicités après la première interruption d'une session.

`rdid=[player_params.rdid]`(Identifiant d'appareil réinitialisable)

L'identifiant publicitaire de l'appareil. Pour les appareils CTV et Smart TV, les acheteurs de publicités programmatiques ont besoin de cet identifiant pour enchérir sur le stock. Sans cela, les taux de remplissage des appareils CTV peuvent être nettement inférieurs.

`msid=[player_params.msid]`(ID de l'application)

L'identifiant de l'application. Cela est par exemple `rdid` requis par les acheteurs de programmes pour le ciblage des appareils CTV et Smart TV.

Pour plus d'informations sur les paramètres de ciblage des appareils pris en charge par GAM, consultez les paramètres de ciblage de [Google Ad Manager](#) sur le site Web de Google.

Erreurs de configuration courantes

Les erreurs de configuration suivantes sont à l'origine des faibles taux de remplissage avec GAM :

- Codé en dur **pmxd** : l'utilisation d'une valeur fixe (par exemple `pmxd=60000`) lorsque les pauses publicitaires sont plus longues (90 secondes, par exemple) laisse le temps restant inutilisé. Utilisez `[session.avail_duration_ms]` à la place.
- **pmad**Valeur faible : si le nombre maximum d'annonces par pod est trop faible, GAM ne pourra pas utiliser toute la durée de la pause, même lorsque l'inventaire est disponible. -1Utilisez-le pour un nombre illimité ou définissez la limite souhaitée.
- Manquant **correlator** — Sans ce paramètre, GAM peut renvoyer des réponses VAST vides pour des pauses publicitaires répétées au cours d'une même session.
- **ppid**Manquant : sans ce paramètre de lecteur, GAM peut cesser de renvoyer des publicités après la première interruption publicitaire d'une session.
- Identifiant d'appareil manquant sur les appareils CTV — Sans `celardid`, les acheteurs programmatiques ne peuvent pas cibler l'appareil, ce qui réduit considérablement le remplissage de l'inventaire CTV.
- Utiliser **vpos=[avail.type]** — MediaTailor ne prend pas en charge la `[avail.type]` variable. Cela se résout en une chaîne vide, ce qui peut entraîner des échecs de demande ou des réponses publicitaires de faible valeur.
- Paramètre du lecteur non transmis lors de l'initialisation de la session : l'URL de votre modèle ADS fait référence à un paramètre du lecteur (tel que `[player_params.ppid]`), mais l'appel d'initialisation de session ne transmet pas la valeur correspondante. GAM reçoit une valeur vide ou nulle pour cette macro, ce qui peut empêcher le remplissage de la demande d'annonce.

Personnalisation du comportement des interruptions de publicité grâce à la suppression des interruptions de publicité

Lorsque vous créez une configuration dans AWS Elemental MediaTailor, vous pouvez spécifier des paramètres de configuration facultatifs qui régissent le comportement des pauses publicitaires, notamment la possibilité de configurer la suppression des pauses publicitaires. Cela vous permet d'adapter les expériences publicitaires à votre contenu vidéo afin de répondre à vos besoins spécifiques.

Restrictions de compatibilité

Vous ne pouvez pas utiliser la suppression des interruptions de publicité dans les cas suivants :

- Workflows VOD et Live-to-VOD. Seuls les flux de travail en direct sont pris en charge.
- Server-guided méthodes d'insertion de publicités (SGAI). Server-guided les méthodes gèrent différemment la prise de décision publicitaire et ne nécessitent pas de configuration de suppression.

Rubriques

- [Configuration de la suppression des interruptions publicitaires](#)

Configuration de la suppression des interruptions publicitaires

Vous pouvez configurer MediaTailor pour ignorer la personnalisation des pauses publicitaires pour le contenu en direct. C'est ce que l'on appelle la suppression des interruptions publicitaires ou la suppression des avantages. Cette rubrique explique comment procéder et explique également comment fonctionne la configuration de la suppression des interruptions de publicité.

La suppression des interruptions publicitaires peut être utilisée dans les cas d'utilisation suivants :

- Grande fenêtre de recherche de manifeste : si un utilisateur démarre la lecture à la limite d'un manifeste mais que la fenêtre de consultation est grande, vous ne pouvez insérer des publicités qu'une fois que l'utilisateur a commencé la lecture. Vous pouvez également insérer des publicités dans une partie de la fenêtre du manifeste. Vous pouvez configurer la suppression des publicités de manière MediaTailor à personnaliser les interruptions publicitaires pendant ou pendant une période spécifiée après le Live Edge.

- Mid-break rejoindre — Si le spectateur commence à regarder un flux vidéo en direct au milieu d'une pause publicitaire, il est probable qu'il changera de chaîne et ne regardera pas la publicité. Grâce à la suppression des publicités, vous pouvez ignorer la personnalisation des interruptions publicitaires si celles-ci ont commencé avant que le spectateur n'ait rejoint le stream.

Configuration de la suppression des publicités

Pour utiliser la suppression des publicités, vous devez configurer un mode de suppression des avantages, une valeur de suppression des avantages et une politique de remplissage de la suppression des avantages de la manière suivante :

- Dans la MediaTailor console
- En utilisant le AWS Command Line Interface (AWS CLI)
- À l'aide de l' MediaTailor API ou en tant que paramètres dans la demande de session de lecture de votre client

Pour plus d'informations sur la configuration à l'aide de paramètres, consultez [Configuration des paramètres de la suppression des publicités — demande de session de lecture](#).

Paramètres de configuration de suppression des publicités

Vous pouvez choisir d'activer ou de désactiver la suppression des publicités. Si vous activez la suppression des publicités, vous indiquez si cette suppression a lieu après le bord de diffusion en direct ou avant le bord de diffusion en direct d'une diffusion en direct. Dans les deux cas, vous spécifiez également une heure, par rapport au Live Edge, pendant laquelle les publicités MediaTailor ne sont pas personnalisées. Lorsque vous activez la suppression de disponibilité, vous pouvez définir une politique de suppression de disponibilité qui s'applique aux remplissages partiels MediaTailor des interruptions de publicité lorsqu'une session démarre en milieu de pause.

Les paramètres de configuration de la suppression de publicités sont les suivants :

- Mode de suppression des annonces : définit le mode de suppression des publicités. Par défaut, la suppression des publicités est désactivée. Valeurs acceptées : OFFBEHIND_LIVE_EDGE, ou AFTER_LIVE_EDGE.
- OFF: Il n'y a pas de suppression des publicités et permet de MediaTailor personnaliser tous les breaks publicitaires.

- **BEHIND_LIVE_EDGE**: MediaTailor ne personnalise pas les pauses publicitaires qui commencent avant le Live Edge, sans la valeur de suppression d'Avail. Cela affecte l'ensemble de la pause publicitaire, et pas seulement les utilisations individuelles des annonces.
- **AFTER_LIVE_EDGE**: MediaTailor ne personnalise pas les pauses publicitaires situées dans le Live Edge, plus la valeur de suppression d'Avail. Cela peut être configuré pour affecter l'intégralité des pauses publicitaires ou pour permettre le remplissage partiel des disponibilités publicitaires.
- Valeur de suppression des annonces : durée par rapport à la limite d'un flux en direct. Valeur acceptée : durée en HH:MM:SS.
- Politique de remplissage de la suppression d'Avail — Définit la politique que MediaTailor s'applique au mode de suppression d'Avail. Valeurs acceptées : **PARTIAL_AVAIL**, **FULL_AVAIL_ONLY**.
 - **BEHIND_LIVE_EDGE**Le mode utilise toujours la politique de **FULL_AVAIL_ONLY** suppression.
 - **AFTER_LIVE_EDGE**Le mode peut être utilisé pour invoquer le remplissage des pauses **PARTIAL_AVAIL** publicitaires lorsqu'une session commence en milieu de pause.

Exemples de paramètres de suppression des publicités

La manière dont les [paramètres de configuration de suppression des publicités interagissent les uns avec les autres](#) vous permet de définir différentes manières de gérer la suppression des publicités et d'utiliser le remplissage avant, pendant ou après la fin de la diffusion en direct de la diffusion en direct. Cette section fournit des exemples illustrant certaines de ces interactions. Utilisez ces exemples pour vous aider à configurer les paramètres de configuration adaptés à votre situation particulière.

Voici des exemples de paramètres de suppression des publicités :

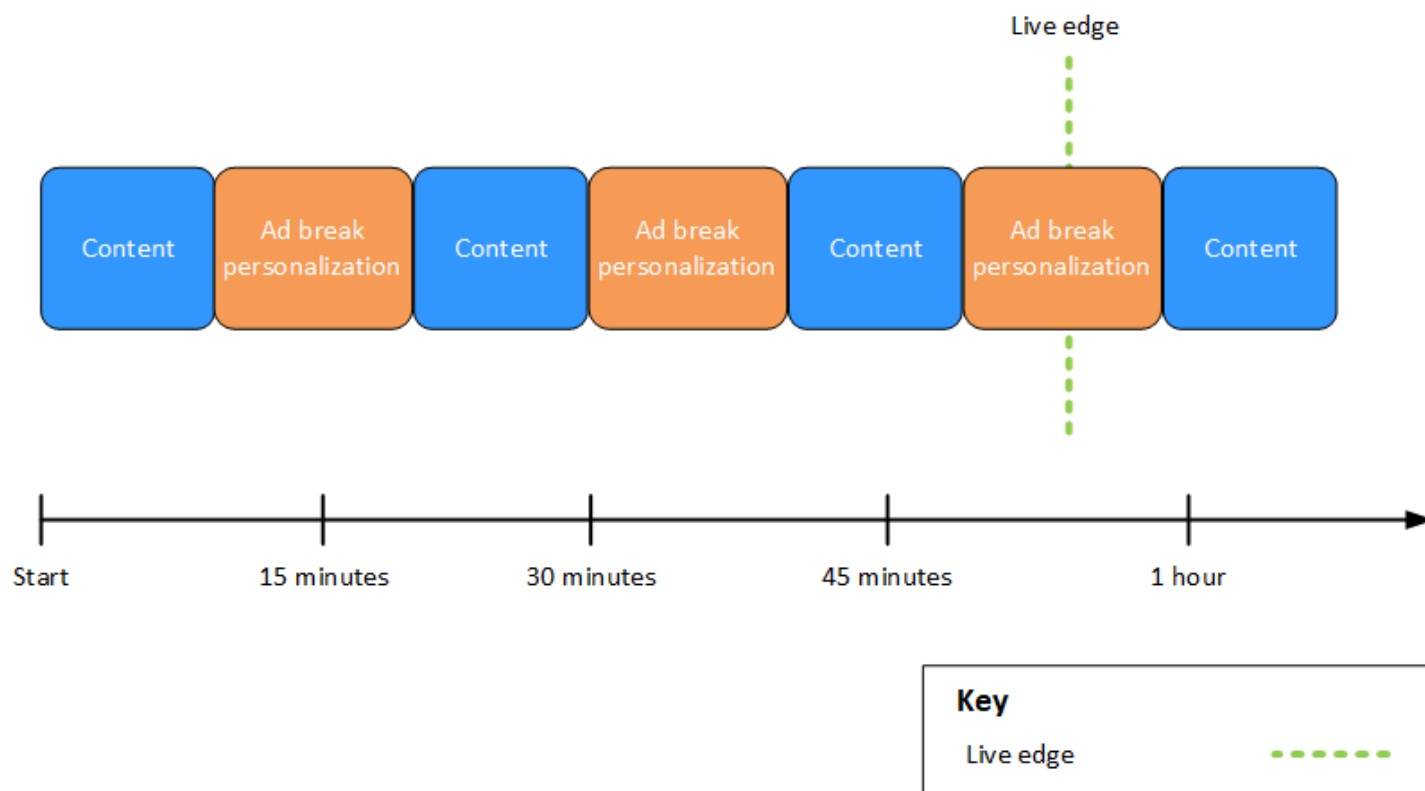
Exemple 1 : pas de suppression de publicités

Lorsque le mode de suppression Avail est activé **OFF**, il n'y a pas de suppression des publicités et permet de MediaTailor personnaliser tous les breaks publicitaires.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct ou d'une pause publicitaire personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Deux pauses publicitaires ont lieu avant le Live Edge, et un autre break publicitaire est en cours pendant le Live Edge. Comme le montre la figure, lorsque le mode de suppression des avantages est activé **OFF**, il MediaTailor

personnalise toutes les interruptions de publicité qui se produisent avant la limite en direct sur la chronologie. MediaTailor personnalise également la pause publicitaire en cours sur le Live Edge.

Avail suppression mode (default): OFF



Exemple 2 : Suppression des publicités par **BEHIND_LIVE_EDGE** avec une valeur synchronisée avec Live Edge

Lorsque le mode de suppression des avantages est défini sur **BEHIND_LIVE_EDGE** et que la valeur de suppression des avantages est définie sur `00:00:00`, la valeur de suppression des avantages est synchronisée avec le Live Edge. MediaTailor ne personnalise pas les pauses publicitaires qui commencent pendant ou avant le Live Edge.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct, d'une pause publicitaire personnalisée ou d'une pause publicitaire non personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur `00:00:00`, chevauche la ligne pointillée du bord actif. Deux interruptions de publicité ont lieu avant le Live Edge, et un autre break de publicité

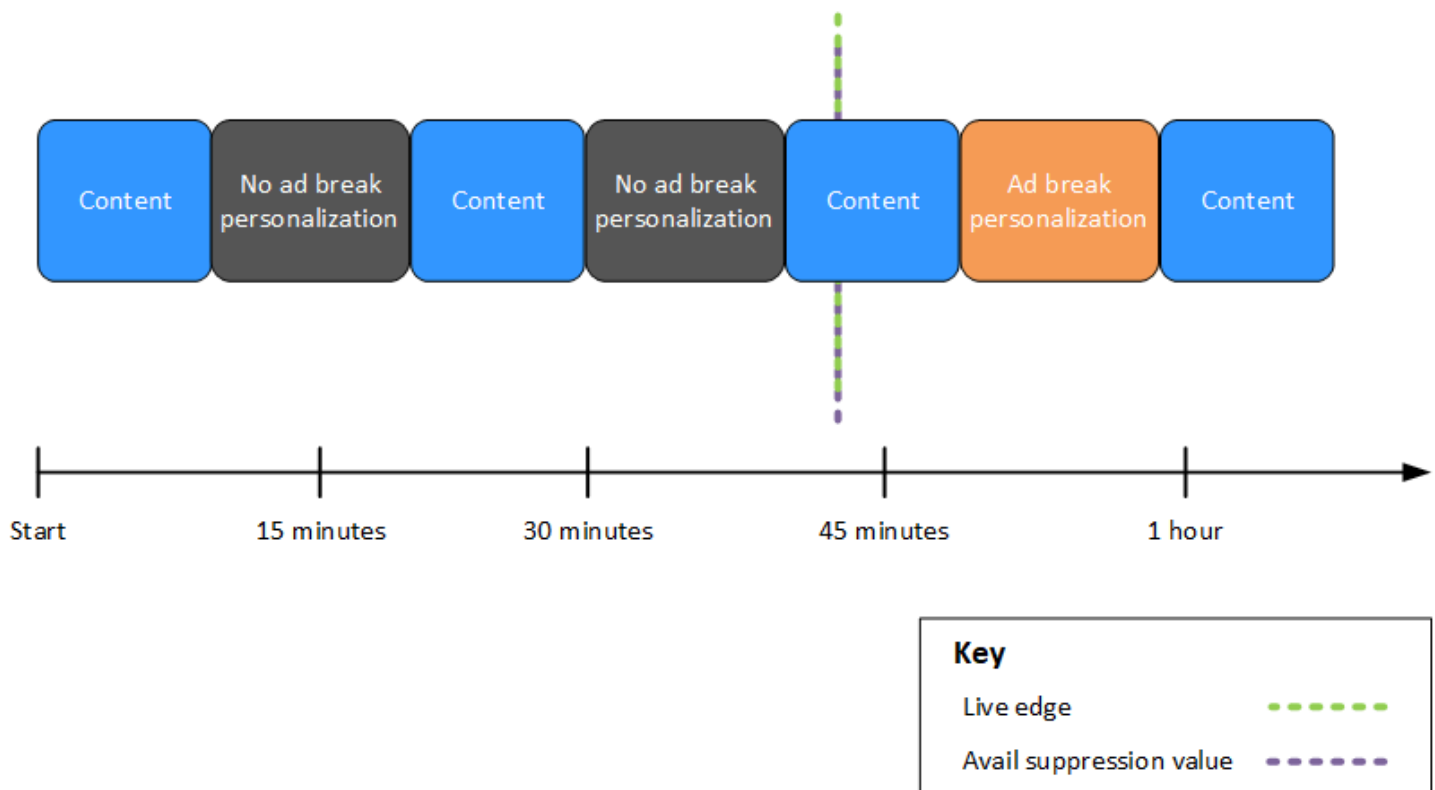
se produit après le Live Edge. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur et que la valeur de suppression des avantages est définie de 00:00:00 manière à ce qu'il soit synchronisé avec le Live Edge, aucune pause publicitaire

MediaTailor ne se produit avant le Live Edge sur la chronologie. BEHIND_LIVE_EDGE

MediaTailor personnalise la pause publicitaire qui se produit après le Live Edge.

Avail suppression mode: **BEHIND_LIVE_EDGE**

Avail suppression value: **00:00:00**



Exemple 3 : Suppression des publicités par **BEHIND_LIVE_EDGE** avec une valeur ajoutée derrière Live Edge

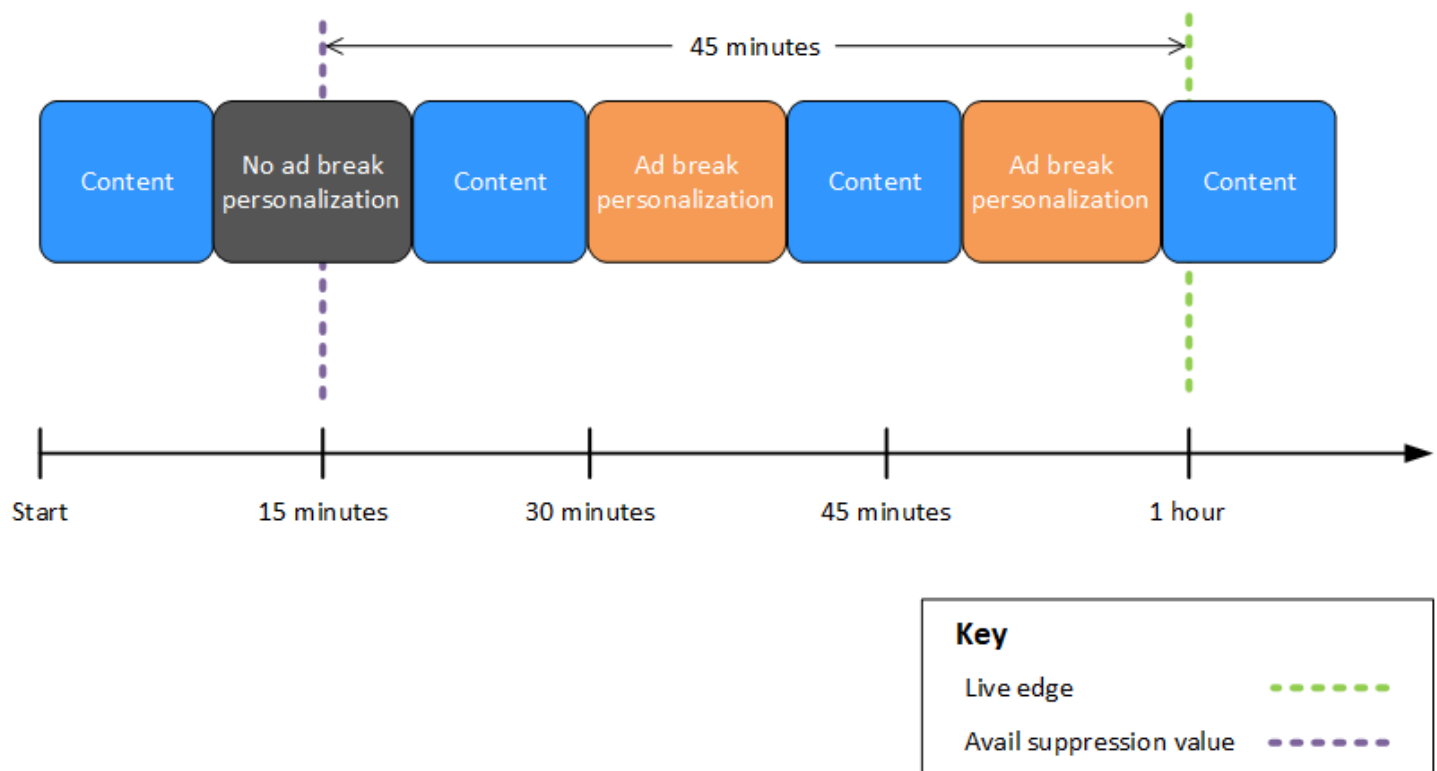
Lorsque le mode de suppression des avantages est défini sur **BEHIND_LIVE_EDGE**, MediaTailor il ne personnalise aucune pause publicitaire à cette date ou avant cette date. Dans cet exemple, MediaTailor personnalise les pauses publicitaires qui commencent jusqu'à 45 minutes après le Live Edge. MediaTailor ne personnalise pas les pauses publicitaires qui commencent à plus de 45 minutes après le Live Edge.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant

laquelle le contenu de la diffusion en direct, d'une pause publicitaire personnalisée ou d'une pause publicitaire non personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur `00:45:00`, apparaît 45 minutes plus tôt dans la chronologie par rapport à la ligne pointillée du Live Edge. La période de 45 minutes entre les lignes pointillées représente la période de suppression des avantages. Une pause publicitaire est en cours au début de la période de suppression des avantages. Deux autres interruptions publicitaires se produisent pendant la période de suppression des avantages. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur `BEHIND_LIVE_EDGE` et que la valeur de suppression des avantages est définie sur « `00:45:00` Derrière le Live Edge », les interruptions de publicité survenant pendant la période de suppression des fonctionnalités sont MediaTailor personnalisées. `BEHIND_LIVE_EDGE` MediaTailor ne personnalise pas la pause publicitaire en cours au début de la période de suppression des avantages.

Avail suppression mode: **BEHIND_LIVE_EDGE**

Avail suppression value: **00:45:00**



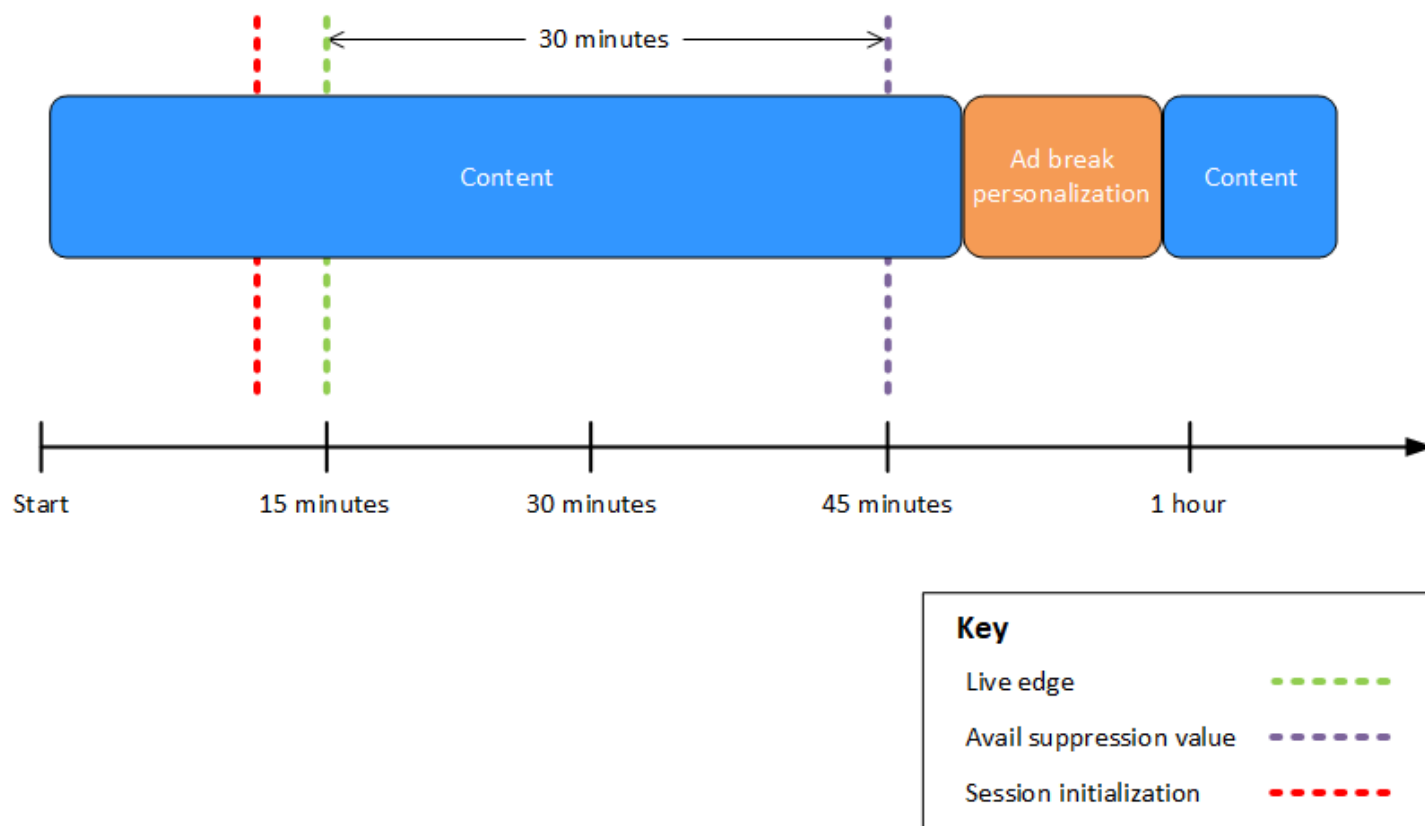
Exemple 4 : Suppression des publicités **AFTER_LIVE_EDGE** sans interruption publicitaire pendant la période de suppression des avantages

Lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE** et que la valeur de suppression des avantages est supérieure à zéro, aucune pause publicitaire MediaTailor n'est personnalisée tant que le temps écoulé de la session n'a pas atteint cette valeur.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct ou d'une pause publicitaire personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur `00:30:00`, apparaît 30 minutes plus tard dans la chronologie par rapport à la ligne pointillée correspondant à la limite active. Une troisième ligne en pointillés, représentant l'initialisation de la session, apparaît plus tôt dans la chronologie par rapport à la ligne pointillée correspondant à l'arête active. La période de 30 minutes entre l'heure de pointe et l'heure de suppression de disponibilité représente la période de suppression de disponibilité. Une interruption publicitaire se produit après la période de suppression des avantages. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE**, la valeur de suppression des avantages est définie sur `00:30:00` après le Live Edge, et l'initialisation de la session a lieu avant le Live Edge, MediaTailor personnalise les interruptions de publicité survenant après la période de suppression des avantages.

Avail suppression mode: **AFTER_LIVE_EDGE**

Avail suppression value: **00:30:00**



Example 5 : Suppression des publicités **AFTER_LIVE_EDGE**
avec politique de remplissage **PARTIAL_AVAIL** et interruption de
publicité en cours à la fin de la période de suppression des publicités

Lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE** et que la valeur de suppression des avantages est supérieure à zéro, aucune pause publicitaire MediaTailor n'est personnalisée tant que le temps écoulé de la session n'a pas atteint cette valeur.

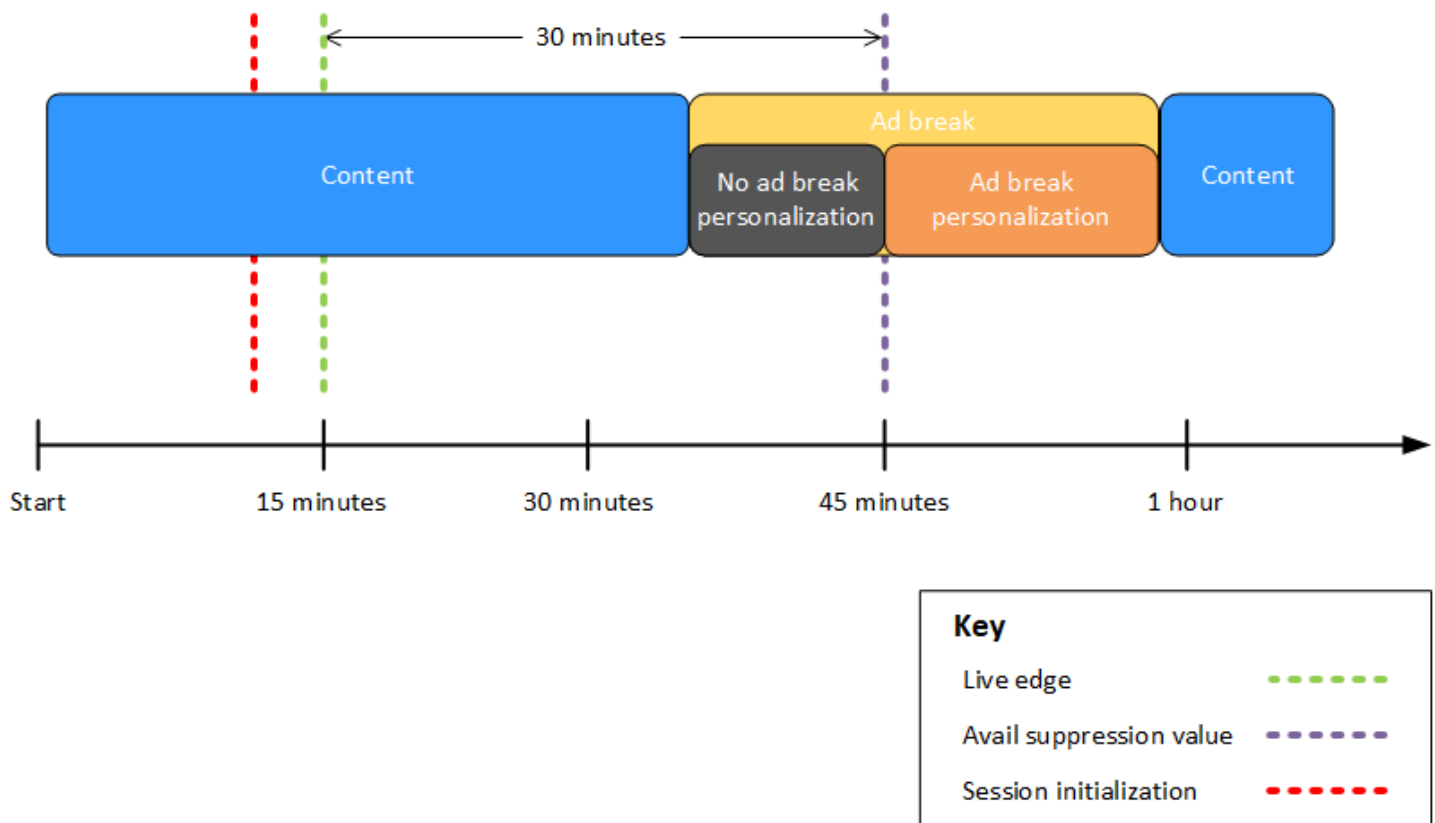
Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct, d'une pause publicitaire personnalisée ou d'une pause publicitaire non personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur **00:30:00**, apparaît 30 minutes plus tard dans la chronologie par rapport à la ligne

pointillée correspondant à la limite active. Une troisième ligne en pointillés, représentant l'initialisation de la session, apparaît plus tôt dans la chronologie par rapport à la ligne pointillée correspondant à l'arrêt actif. La période de 30 minutes entre l'heure de pointe et l'heure de suppression de disponibilité représente la période de suppression de disponibilité. Une pause publicitaire est en cours à la fin de la période de suppression des avantages. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur `AFTER_LIVE_EDGE`, la valeur de suppression des avantages est définie sur `00:30:00` après le Live Edge, la politique de remplissage de la suppression des avantages est définie sur `PARTIAL_AVAIL`, et l'initialisation de la session a lieu avant le Live Edge, MediaTailor personnalise les interruptions publicitaires survenant après la période de suppression des avantages. Pour la pause publicitaire en cours à la fin de la période de suppression des avantages, MediaTailor personnalise la partie de cette interruption publicitaire qui se produit après la période de suppression des avantages, mais ne personnalise pas la partie de cette interruption publicitaire qui se produit pendant la période de suppression des avantages.

Avail suppression mode: `AFTER_LIVE_EDGE`

Avail suppression value: `00:30:00`

Avail suppression fill policy: `PARTIAL_AVAIL`



Exemple 6 : Suppression des publicités **AFTER_LIVE_EDGE** avec politique de remplissage **PARTIAL_AVAIL** et interruption de publicité en cours entre la date d'initialisation de la session et la fin de la période de suppression des publicités

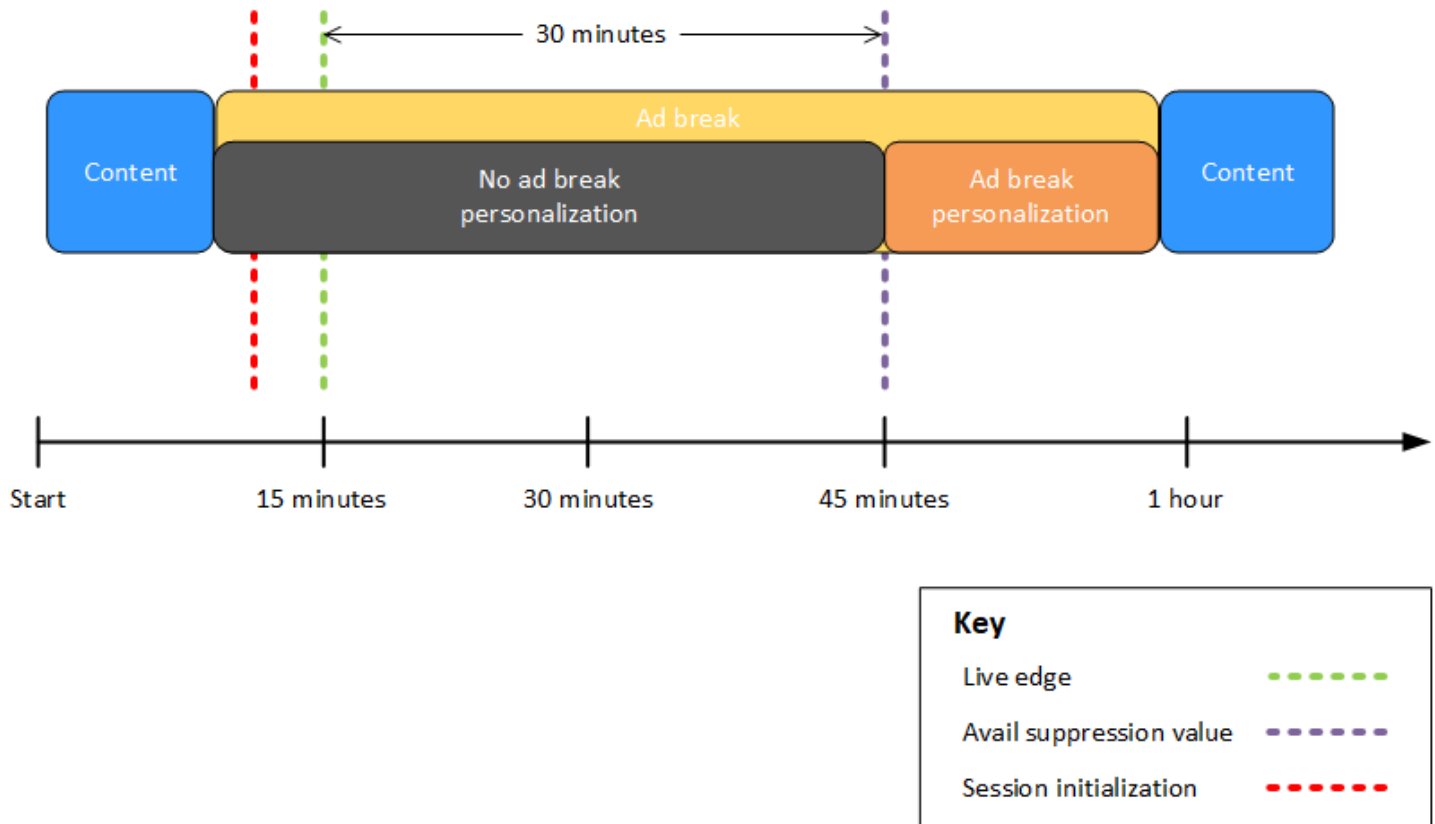
Lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE** et que la valeur de suppression des avantages est supérieure à zéro, aucune pause publicitaire MediaTailor n'est personnalisée tant que le temps écoulé de la session n'a pas atteint cette valeur.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct, d'une pause publicitaire personnalisée ou d'une pause publicitaire non personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur `00:30:00`, apparaît 30 minutes plus tard dans la chronologie par rapport à la ligne pointillée correspondant à la limite active. Une troisième ligne en pointillés, représentant l'initialisation de la session, apparaît plus tôt dans la chronologie par rapport à la ligne pointillée correspondant à l'arête active. La période de 30 minutes entre l'heure de pointe et l'heure de suppression de disponibilité représente la période de suppression de disponibilité. Une pause publicitaire est en cours entre une période antérieure à l'initialisation de la session et une période postérieure à la période de suppression des avantages. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE**, la valeur de suppression des avantages est définie sur `00:30:00` après le Live Edge, la politique de remplissage de la suppression des avantages est définie sur **PARTIAL_AVAIL**, et l'initialisation de la session a lieu avant le Live Edge, MediaTailor personnalise les interruptions publicitaires survenant après la période de suppression des avantages. Pour la pause publicitaire en cours avant, pendant et après la période de suppression des avantages, MediaTailor personnalise la partie de cette interruption publicitaire qui se produit après la période de suppression des avantages, mais ne personnalise pas la partie de cette interruption publicitaire qui se produit avant ou pendant la période de suppression des avantages.

Avail suppression mode: **AFTER_LIVE_EDGE**

Avail suppression value: **00:30:00**

Avail suppression fill policy: **PARTIAL_AVAIL**



Example 7 : Suppression des publicités **AFTER_LIVE_EDGE** avec une interruption publicitaire en cours au début de la période de suppression des avantages

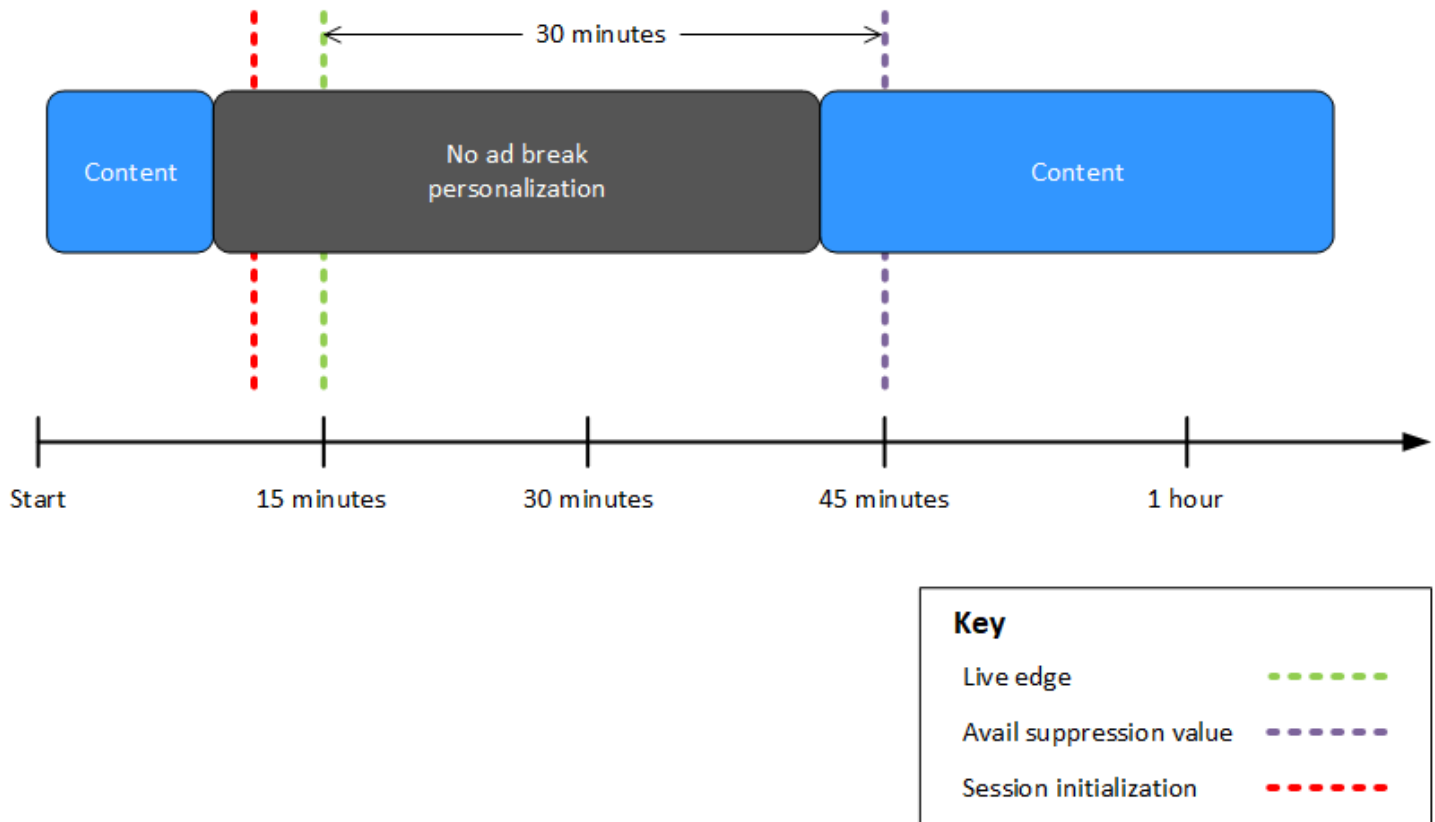
Lorsque le mode de suppression des avantages est défini sur **AFTER_LIVE_EDGE** et que la valeur de suppression des avantages est supérieure à zéro, aucune pause publicitaire MediaTailor n'est personnalisée tant que le temps écoulé de la session n'a pas atteint cette valeur.

Dans la figure suivante, différents blocs sont disposés horizontalement le long d'une chronologie qui progresse de gauche à droite. Chaque bloc représente une partie du temps pendant laquelle le contenu de la diffusion en direct ou d'une pause publicitaire non personnalisée est diffusé. Une ligne en pointillés représente le bord en direct actuel de la diffusion en direct. Une autre ligne en pointillés, représentant la valeur de suppression d'utilisation définie sur **00:30:00**, apparaît 30 minutes plus tard dans la chronologie par rapport à la ligne pointillée correspondant à la limite active. Une troisième ligne en pointillés, représentant l'initialisation de la session, apparaît plus

tôt dans la chronologie par rapport à la ligne pointillée correspondant à l'arête active. La période de 30 minutes entre l'heure de pointe et l'heure de suppression de disponibilité représente la période de suppression de disponibilité. Une pause publicitaire est en cours entre une période antérieure à l'initialisation de la session et une période comprise entre la période de suppression des fonctionnalités. Comme le montre la figure, lorsque le mode de suppression des avantages est défini sur `AFTER_LIVE_EDGE`, la valeur de suppression des avantages est définie sur `00:30:00` après le live edge, et l'initialisation de la session a lieu avant l'heure du live-edge, mais après le début de la pause publicitaire, MediaTailor cela ne personnalise pas cette pause publicitaire.

Avail suppression mode: `AFTER_LIVE_EDGE`

Avail suppression value: `00:30:00`



Configuration des paramètres de la suppression des publicités — demande de session de lecture

Vous pouvez configurer les paramètres de suppression des publicités via les paramètres figurant dans votre demande initiale de session de lecture côté serveur ou côté client auprès de MediaTailor. Si vous avez déjà configuré les paramètres de suppression des publicités via la MediaTailor console ou AWS Elemental MediaTailor l'API, ces paramètres remplacent ces paramètres.

Le mode et la valeur de suppression des annonces sont requis pour que la suppression fonctionne. Ces paramètres ne peuvent pas être configurés à partir de sources différentes. Par exemple, vous ne pouvez pas configurer un paramètre avec la MediaTailor console et un autre avec un paramètre de requête.

MediaTailor prend en charge les paramètres de suppression des publicités suivants.

Nom	Description	Valeurs acceptées
<code>availSuppressionMode</code>	Définit le mode de suppression des publicités. Par défaut, la suppression des publicités est OFF. Lorsqu'il est défini sur <code>BEHIND_LIVE_EDGE</code> , MediaTailor il ne remplit pas les interstices publicitaires pendant ou en <code>aws.availSuppressionValue</code> retard. Lorsqu'il est défini sur <code>AFTER_LIVE_EDGE</code> , MediaTailor il ne remplit pas les interstices publicitaires pendant ou après la période de suppression des avantages. La période de suppression des avantages s'étend du moment de la mise en ligne à l'heure initiale, plus le <code>aws.availSuppressionValue</code> temps de mise en mémoire tampon supplémentaire.	• OFF • <code>BEHIND_LIVE_EDGE</code> • <code>AFTER_LIVE_EDGE</code>
<code>availSuppressionValue</code>	Durée par rapport à la limite d'un flux en direct.	Un code UTF-8 URL-encodé temporel est entré <code>HH:MM:SS</code> . Par exemple, 1 heure et 30 minutes est <code>01%3A30%3A00</code> .
<code>availSuppressionFullPolicy</code>	Définit la politique à appliquer au mode de suppression des avantages. <code>BEHIND_LIVE_EDGE</code> utilise toujours la politique de suppression complète de la disponibilité. <code>AFTER_LIVE_EDGE</code> peut être utilisé pour invoquer des remplissages partiels lorsqu'une session commence en milieu de pause publicitaire.	• <code>PARTIAL_AVAAIL</code> - non disponible pour le mode de <code>BEFORE_LIVE_EDGE</code>

Nom	Description	Valeurs acceptées
		VE_EDGE suppression • FULL_AVAI L_ONLY - la valeur par défaut pour le mode de AFTER_LIV E_EDGE suppression

Server-side configuration

Le paramètre de requête de base est `aws.availSuppression`, suivi de paires nom-valeur de paramètre facultatives. Pour créer la requête, ajoutez `aws.availSuppression=` à la fin de la session de lecture la demande de MediaTailor, suivie des noms et valeurs des paramètres. Pour plus d'informations sur la création d'une demande de session de lecture côté serveur, consultez [MediaTailor suivi des publicités et rapports côté serveur](#).

Exemple : HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?
aws.availSuppressionMode=BEHIND_LIVE_EDGE&aws.availSuppressionValue=00%3A00%3A21
```

Server-side la syntaxe de la requête est répertoriée dans le tableau suivant.

Composant de la chaîne de requête	Description
?	Caractère restreint qui marque le début d'une requête.
aws .	Requête de base, suivie de paramètres constitués de paires nom-valeur. Pour obtenir la liste de tous les paramètres disponibles, consultez Configuration des paramètres de la suppression des publicités — demande de session de lecture .

Composant de la chaîne de requête	Description
=	Associe le nom du paramètre à une valeur. Par exemple, <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE</code> .
&	Concatène les paramètres de la requête. Par exemple, <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE &aws.availSuppressionValue= 00:30:00&aws.availSuppressionFillPolicy= FULL_AVAIL_ONLY ></code> .

Client-side configuration

Incluez `availSuppression` des paramètres dans la requête POST de votre client à MediaTailor. Pour plus d'informations sur la création d'une demande de session de lecture côté client, consultez [Client-side suivi des publicités](#).

Exemple : HLS

```
POST parent.m3u8
{
  "availSuppression": {
    "mode": "BEHIND_LIVE_EDGE",
    "value": "00:00:21",
    "fillPolicy": "FULL_AVAIL_ONLY"
  }
}
```

MediaTailor pare-chocs et insertion

Les bumpers sont de courts clips vidéo ou audio que vous ne pouvez pas ignorer et qui sont diffusés au début ou avant la fin d'une pause publicitaire. AWS Elemental MediaTailor

Les conditions suivantes s'appliquent aux pare-chocs :

- Les pare-chocs doivent durer 10 secondes ou moins.
- Les pare-chocs peuvent être insérés au début d'un break publicitaire, juste avant la fin d'un break publicitaire, ou les deux.

- Les bumpers sont diffusés pendant chaque pause publicitaire d'une session de diffusion, sauf si le pré-roll est configuré. Si le pré-lancement est configuré, les pare-chocs ne joueront pas pendant la pause préalable au lancement. Au lieu de cela, ils joueront à chaque pause suivante après le pré-roll.
- Pour HLS, vous devez inclure l'attribut `duration` dans chaque SCTE-35 EXT-X-CUE-OUT balise.
- Les bumpers sont transcodés pour correspondre au contenu source.
- Les pare-chocs ne vous sont pas facturés.

Configuration des pare-chocs

Pour utiliser les pare-chocs, configurez les URL des pare-chocs avec la MediaTailor console, l'API MediaTailor ou le AWS Command Line Interface (AWS CLI). Vous pouvez configurer un pare-choc de départ, un pare-choc de fin ou les deux. Les pare-chocs sont stockés sur un serveur, tel qu'Amazon Simple Storage Service (Amazon S3). Les URL des pare-chocs indiquent l'emplacement du ou des actifs du pare-choc stockés.

Exemples d'URL de pare-chocs de début et de fin :

URL du pare-choc de démarrage : `https://s3.amazonaws.com/startbumperad`

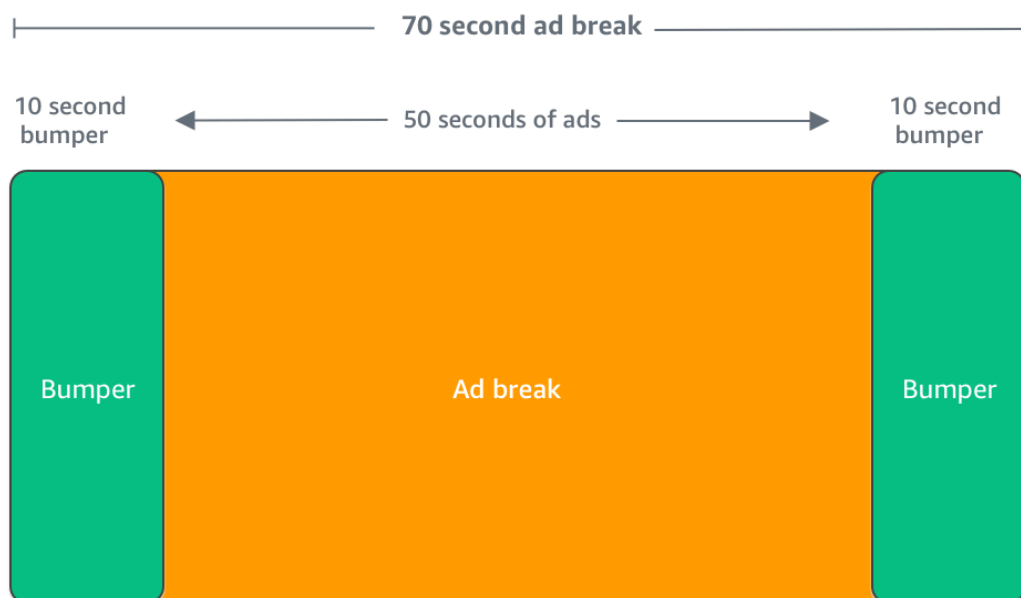
URL du pare-choc de fin : `https://s3.amazonaws.com/endbumperad`

Exemple

Voici un exemple de comportement publicitaire exceptionnel.

Exemple 1 : pare-chocs de début et d'arrivée

Dans cet exemple, les pare-chocs de début et de fin sont activés. Le serveur de décision publicitaire dispose de 50 secondes de publicités personnalisées pour combler une pause publicitaire de 70 secondes. Le pare-choc de 10 secondes s'affiche au début du break publicitaire, 50 secondes de diffusion des publicités, puis le pare-choc de fin de 10 secondes.



MediaTailor pré-enroulement et insertion

Note

Les publicités pré-roll configurables ne sont disponibles que pour les flux de travail en direct. Pour en savoir plus sur le fonctionnement de l'insertion publicitaire (y compris le pré-roll) pour la VOD, consultez [Comportement d'assemblage publicitaire pour la VOD](#).

AWS Elemental MediaTailor peut insérer des publicités au début d'une session de lecture, avant le début du contenu principal. Il s'agit de publicités preroll.

Pour insérer des publicités preroll, renseignez les champs Live pre-roll ad decision server (Serveur ADS de publicités preroll en direct) et Live pre-roll maximum allowed duration (Durée maximale autorisée pour le preroll en direct) dans les paramètres Additional (Supplémentaires) de votre configuration, comme décrit à la section [Paramètres de configuration facultatifs](#).

1. Lorsqu'il MediaTailor reçoit une demande de diffusion, il envoie une demande à l'ADS pour des publicités pré-roll sur la base des champs suivants de la configuration de MediaTailor diffusion :
 - Le serveur de décision publicitaire pré-roll en direct est l'URL du serveur de décision publicitaire (ADS) à laquelle MediaTailor envoie la demande d'annonces pré-roll.
 - La durée maximale autorisée avant la diffusion en direct est la durée maximale totale des publicités prédiffusées. MediaTailor effectue l'action suivante en fonction de la durée maximale autorisée :
 - Si la durée totale des annonces dans la réponse ADS est inférieure à la valeur que vous avez indiquée dans la durée maximale autorisée du pré-lancement en direct, MediaTailor insère toutes les annonces. Lorsque la dernière annonce est terminée, elle revient MediaTailor immédiatement au contenu sous-jacent.
 - Si la durée totale des annonces dans la réponse ADS est supérieure à la valeur que vous avez indiquée dans la durée maximale autorisée pour la pré-diffusion en direct MediaTailor , sélectionnez un ensemble d'annonces correspondant à la durée sans dépasser la durée. MediaTailor insère ces publicités sans les couper ni les tronquer. MediaTailor revient au contenu sous-jacent lorsque la dernière annonce sélectionnée est terminée.
2. Lorsqu'il MediaTailor reçoit la réponse pré-roll de l'ADS, il manipule le manifeste pour ajouter des liens vers les annonces pré-roll. MediaTailor calcule l'heure de début de la pause publicitaire préalable à la diffusion de la publicité comme suit :
 - Pour DASH, la formule est $(publishTime - availabilityStartTime) - \max(suggestedPresentationDelay, minBufferTime)$.
 - Pour HLS, la formule est $\max(2 * EXT-X-TARGETDURATION, EXT-X-START:TIMEOFFSET)$.
3. MediaTailor détermine les mesures à prendre pour les pauses publicitaires qui ne sont pas des pré-lancements. Si le pré-roll chevauche un autre break publicitaire, MediaTailor cela ne personnalise pas la partie superposée du break publicitaire.

MediaTailor ardoise et insertion

Note

Slate n'est disponible que pour le live et les live-to-VOD flux de travail.

Avec AWS Elemental MediaTailor, vous pouvez désigner une ardoise pour les pauses publicitaires. Une ardoise est une MP4 ressource par défaut qui est insérée dans un flux, telle qu'une image fixe ou une vidéo en boucle, qui est diffusée à la place du contenu en direct.

AWS Elemental MediaTailor affiche Slate pendant les pauses publicitaires dans les situations spécifiques suivantes :

- Pour combler le temps restant dans une pause qui n'a pas été remplacé par des publicités personnalisées
- Si le serveur ADS (Ad Decision Server) répond par une réponse VAST ou VMAP vide
- Pour les conditions d'erreur, telles que le délai d'expiration de l'ADS ou une erreur HTTP 500 provenant de l'ADS
- Si une annonce ne peut pas être insérée par MediaTailor (par exemple si le transcodage n'est pas terminé)

Si vous ne configurez pas de liste, MediaTailor utilise par défaut le flux de contenu sous-jacent lorsque l'une des conditions ci-dessus est remplie.

Configuration de l'ardoise

Vous désignez l'ardoise dans le volet de configuration supplémentaire de la [MediaTailor console](#). MediaTailor télécharge la liste à partir de l'URL que vous spécifiez et la transcode dans les mêmes formats que votre contenu. Vous pouvez contrôler la durée maximale pendant laquelle une liste sera affichée grâce à la configuration optionnelle du seuil de personnalisation dans la MediaTailor console. Pour de plus amples informations, veuillez consulter [the section called “Comment fonctionne le seuil de personnalisation”](#).

L'ardoise doit être un MP4 actif de haute qualité contenant à la fois du son et de la vidéo. La configuration de la liste est facultative pour les configurations non VPAID, mais obligatoire pour les flux de travail VPAID.

Note

Si le serveur qui héberge votre liste utilise HTTPS, son certificat doit provenir d'une autorité de certification reconnue. Il ne peut pas s'agir d'un certificat auto-signé. Si vous utilisez un certificat auto-signé, vous ne pouvez pas récupérer et assembler l'ardoise dans les manifestes à partir de l'origine du contenu.

Comment fonctionne le seuil de personnalisation

Le seuil de personnalisation définit la durée maximale pendant laquelle une annonce n'est pas utilisée (en secondes) pendant une pause publicitaire. Cette fonctionnalité s'applique spécifiquement au remplacement des publicités dans les flux en direct et VOD, plutôt qu'à l'insertion d'annonces, car elle repose sur un flux de contenu sous-jacent.

Le comportement varie en fonction de trois scénarios :

1. Lorsque la personnalisation est désactivée :
 - a. L'ardoise sera insérée pendant toute la durée du temps non rempli
 - b. Les pare-chocs de début et de fin seront insérés une fois configurés (pour plus d'informations, voir) [MediaTailor pare-chocs et insertion](#)
 - c. Les publicités seront insérées normalement
2. Lorsque la personnalisation est activée et que le seuil est inférieur à la durée de la pause :
 - a. Si le temps non occupé dépasse le seuil de personnalisation :
 - i. MediaTailor abandonne la personnalisation du break publicitaire
 - ii. Le contenu sous-jacent est affiché
 - iii. Aucune publicité, aucune ardoise ou aucun pare-chocs n'est inséré
 - b. Si le temps libre est inférieur au seuil de personnalisation :
 - i. Les annonces et l'ardoise sont insérées
 - ii. Les pare-chocs sont insérés s'ils sont configurés
3. Lorsque la personnalisation est activée et que le seuil est supérieur à la durée de la pause :
 - a. Des publicités seront insérées
 - b. L'ardoise sera insérée pendant le temps restant dans le break publicitaire
 - c. Des pare-chocs seront insérés s'ils sont configurés

Le tableau suivant fournit des comportements détaillés pour des scénarios spécifiques dans le cadre des protocoles HLS et DASH :

Matrice de comportement détaillée

Scénario	Personnalisation désactivée	Personnalisation activée pendant une durée inférieure à la durée de la pause	Personnalisation activée au-delà de la durée de la pause
VAST ou VMAP vide	Ardoise insérée	Aucune ardoise insérée	Ardoise insérée
Délai d'expiration de l'ADS	Ardoise insérée	Aucune ardoise insérée	Ardoise insérée
L'annonce n'est pas disponible (Vast 404)	Ardoise insérée	Aucune ardoise insérée	Ardoise insérée
La durée des annonces est plus longue que la pause publicitaire	Annonces insérées	Aucune annonce insérée	Annonces insérées
Indiquez le temps qui n'est pas entièrement utilisé par une annonce de remplacement	Ardoise insérée	Si le seuil de personnalisation est supérieur au temps non rempli : Non ads/slates inséré, sinon : annonces et ardoises insérées	Ardoise insérée
Preroll avec VAST vide	Ardoise insérée, aucun prérouleau inséré	Aucune ardoise insérée, aucun prérouleau inséré	Ardoise insérée, aucun prérouleau inséré
Prélancement avec délai d'attente ADS	Ardoise insérée, aucun prérouleau inséré	Aucune ardoise insérée, aucun prérouleau inséré	Ardoise insérée, aucun prérouleau inséré

Scénario	Personnalisation désactivée	Personnalisation activée pendant une durée inférieure à la durée de la pause	Personnalisation activée au-delà de la durée de la pause
		preroulean inséré	
Propulser lorsque l'annonce n'est pas disponible (Vast 404)	Ardoise insérée, aucun preroulean inséré	Aucune ardoise insérée, aucun preroulean inséré	Ardoise insérée, aucun preroulean inséré
Pare-chocs avec VAST vide	Ardoise insérée, pare-chocs inséré	Aucune ardoise insérée, aucun pare-chocs inséré	Ardoise insérée, pare-chocs inséré
Pare-chocs avec temporisation ADS	Ardoise insérée, pare-chocs inséré	Aucune ardoise insérée, aucun pare-chocs inséré	Ardoise insérée, pare-chocs inséré
Des pare-chocs lorsque la publicité n'est pas disponible (Vast 404)	Ardoise insérée, pare-chocs inséré	Aucune ardoise insérée, aucun pare-chocs inséré	Ardoise insérée, pare-chocs inséré

Scénario	Personnalisation désactivée	Personnalisation activée pendant une durée inférieure à la durée de la pause	Personnalisation activée au-delà de la durée de la pause
Des pare-chocs pour combler le temps qui n'est pas entièrement utilisé par le remplacement des publicités	Ardoise insérée, pare-chocs inséré et inséré	Si le seuil de personnalisation est supérieur au temps non rempli : Non inséré bumpers/ads/slides inserted, else: No bumpers/ads/slides	Ardoise insérée, pare-chocs inséré et inséré

Considérations importantes

N'oubliez pas les points suivants lorsque vous travaillez avec Slate et le seuil de personnalisation.

- Ce comportement est cohérent entre les protocoles HLS et DASH
- La fonctionnalité de seuil de personnalisation ne s'applique que lorsque Slate est configurée
- Lorsque Slate est configurée et que le seuil de personnalisation n'est pas configuré, Slate sera diffusée soit pendant toute la durée de disponibilité d'origine, soit pendant le temps restant après avoir renseigné les publicités
- Pour les annonces VPAID, MediaTailor insère une ardoise pendant toute la durée de l'annonce VPAID afin de réserver de l'espace pour les publicités que le lecteur vidéo insérera

Configuration de Slate et VPAID

Important

La configuration de Slate est obligatoire lors de l'utilisation de publicités VPAID. MediaTailor insère une ardoise pour contenir de l'espace pour les publicités VPAID que le lecteur vidéo

insérera. La durée de la liste peut être légèrement plus longue que la durée de la publicité VPAID pour tenir compte de l'interactivité des utilisateurs.

Le lecteur vidéo gère ensuite la publicité VPAID en fonction des métadonnées de reporting côté client qui sont renvoyées, comme MediaTailor décrit dans. [Exigences relatives au VPAID](#) Pour plus d'informations sur la création de rapports côté client, consultez [the section called "Client-side suivi"](#).

Transcodage des annonces dans AWS Elemental MediaTailor

AWS Elemental MediaTailor transcode les publicités, les bumpers et les ardoises en fonction du format et des spécifications de vos flux vidéo d'origine, garantissant ainsi une diffusion fluide pendant les pauses publicitaires. MediaTailor prend en charge deux approches de transcodage : le transcodage dynamique (par défaut) et un profil de transcodage personnalisé. Comprendre les différences entre ces approches vous aidera à déterminer la configuration adaptée à vos flux de travail.

Rubriques

- [Transcodage dynamique](#)
- [Profils de transcodage personnalisés](#)

Transcodage dynamique

Le transcodage dynamique est le comportement de transcodage publicitaire par défaut dans. MediaTailor Lorsque'une annonce, un bumper ou une ardoise doit être transcodée, il inspecte MediaTailor automatiquement le manifeste d'origine et détermine les paramètres de transcodage au moment de l'exécution, sans aucune configuration de votre part.

Ce que gère le transcodage dynamique

Le transcodage dynamique gère automatiquement les formulaires HLS et les DASH flux de travail suivants.

- Correspondance de codec standard : MediaTailor correspond au profil de codec du flux d'origine. Notez que la correspondance des niveaux de codec n'est pas garantie.
- Normalisation audio : applique MediaTailor automatiquement la normalisation audio aux publicités transcodées en utilisant un niveau cible par défaut de -24 LKFS. Cela garantit que les niveaux audio

des annonces correspondent à votre contenu d'origine, sans aucune configuration supplémentaire. Si votre flux de travail nécessite une cible de normalisation audio personnalisée autre que la cible par défaut-24 LKFS, un profil de transcodage personnalisé est requis.

- **Correspondance standard de la longueur des segments** : par défaut, MediaTailor transcode les annonces en utilisant des segments de 2 secondes. Si votre joueur a besoin de faire correspondre la longueur des segments, vous pouvez activer le `calculateSegmentLength` drapeau, ce qui permet de MediaTailor calculer automatiquement la longueur de segment correcte en analysant les 10 premiers segments depuis l'origine et en sélectionnant la durée la plus courante comme longueur de segment. Cet indicateur est désactivé par défaut et doit être activé par AWS Support. Contactez AWS Support pour demander l'activation de `calculateSegmentLength` votre compte ou de configurations spécifiques.
- **WebVTT sous-titres standard** : pour les HLS flux de travail, MediaTailor préserve les références aux groupes de sous-titres dans la liste de lecture multivariante personnalisée lorsque des WebVTT sous-titres sont présents dans le manifeste d'origine. Au cours des segments publicitaires, MediaTailor insère WebVTT des segments vides pour assurer la continuité des pistes de sous-titres, car les créations publicitaires n'incluent généralement pas de pistes de sous-titres. Pour les DASH flux de travail, MediaTailor nécessite que les ensembles d'adaptation des sous-titres du manifeste d'origine soient segmentés (comme la vidéo et l'audio). Les périodes publicitaires créées par contiennent MediaTailor uniquement les ensembles d'adaptation fournis par le créateur publicitaire ; étant donné que les créations publicitaires n'incluent généralement pas de pistes de sous-titres, les périodes publicitaires ne contiendront pas de sous-titre. `AdaptationSet` Les périodes de contenu conservent leurs ensembles d'adaptation des sous-titres comme d'habitude.
- **CMAF support de format** : MediaTailor détecte si un flux HLS d'origine utilise CMAF (fMP4) des segments ou des TS segments standard et transcode les publicités pour qu'elles correspondent en conséquence. En effet DASH, CMAF l'emballage est pris en charge de manière transparente, car DASH il utilise intrinsèquement ISO-BMFF (fMP4) comme format de conteneur. Aucune configuration supplémentaire n'est requise pour l'un ou l'autre protocole.

Note

MediaTailor ne prend pas en charge HLS les flux utilisant HVC1 (HEVC) comme codec avec des TS segments. Conformément à la spécification de création HLS pour les appareils Apple, section 1.5, le format de conteneur pour les HEVC vidéos doit être. fMP4 HLS les flux utilisant HEVC doivent donc être utilisés CMAF avec des segments MP4 fragmentés.

- **Personnalisation par trickplay** : transcode MediaTailor automatiquement les publicités avec des variantes de trickplay et des flux d'images associés (y compris les I-frame manifestes, les vignettes et les flux d'images) pour les deux et en respectant les spécifications du contenu d'origine. HLS DASH Un profil de transcodage personnalisé n'est plus requis pour cette fonctionnalité.
- **DASHManifestes compacts** : prend MediaTailor automatiquement en charge les DASH manifestes compacts, qui élèvent l'SegmentTemplateélément AdaptationSet au niveau Representation des éléments individuels, réduisant ainsi la taille globale du manifeste et améliorant l'efficacité de livraison. Un profil de transcodage personnalisé n'est plus requis pour cette fonctionnalité.

Quand le transcodage dynamique risque de ne pas être suffisant

Le transcodage dynamique gère la majorité des scénarios de transcodage publicitaire. Cependant, dans certains cas, une configuration supplémentaire ou un profil de transcodage personnalisé peuvent être nécessaires.

- **Correspondance du débit** — Le transcodage dynamique ne garantit pas une correspondance précise du débit par défaut. Si votre flux de travail nécessite que les débits publicitaires soient étroitement alignés sur les débits d'origine, vous pouvez activer le `enableBitrateRounding` drapeau, qui permet d'arrondir le débit jusqu'à 5 % afin de mieux correspondre aux valeurs d'origine. Cet indicateur doit être activé par AWS Support. Contactez AWS Support pour demander l'activation de `enableBitrateRounding` votre compte ou de configurations spécifiques. Si votre flux de travail nécessite une correspondance de débit supérieure à 5 %, un profil de transcodage personnalisé est requis.
- Une cible de normalisation audio personnalisée autre que la cible MediaTailor par défaut de -24 LKFS
- Non-integer longueurs de segment (par exemple, 1,92 s ou 3,84 s)
- Système audio Dolby Vision
- Valeurs PID PMT, vidéo ou audio spécifiques requises par les systèmes ou lecteurs en aval
- Taux d'échantillonnage audio personnalisés (par exemple, 44,1 kHz)
- Rapports hauteur/largeur en pixels (PAR) personnalisés, par exemple, 1:1 pour les pixels carrés ou 40:33 pour un écran large NTSC D1/DV
- Pistes audio ou pistes de sous-titres nommées spécifiques

Profils de transcodage personnalisés

Un profil de transcodage personnalisé (CTP) est une configuration de MediaConvert tâche AWS Elemental créée explicitement auprès de laquelle vous vous inscrivez. MediaTailor Lorsque'un CTP est associé à une MediaTailor configuration, il MediaTailor utilise les paramètres que vous avez définis pour transcoder chaque annonce, chaque bumper et chaque ardoise, ce qui vous permet de contrôler précisément les codecs, les débits, les niveaux de normalisation audio, la longueur des segments, les valeurs PID, les fréquences d'échantillonnage, etc. Étant donné qu'un CTP remplace complètement le transcodage dynamique, vous devez configurer explicitement toutes les fonctionnalités dont vous avez besoin, y compris les fonctionnalités astucieuses (I-frame manifestes, miniatures et flux d'images), dans le profil.

Un CTP est conçu pour correspondre à une configuration d'origine spécifique. Si le contenu source diffère de la configuration pour laquelle le CTP a été créé à l'origine, les publicités risquent de ne pas être correctement transcodées en raison d'une NO_VARIANT_MATCH erreur. Cela se produit lorsque les sorties CTP ne s'alignent pas sur les variantes présentes dans le manifeste d'origine. Cela s'applique à la fois au live et aux VOD flux de travail. Par exemple, VOD les bibliothèques qui contiennent du contenu provenant de différentes sources ou codé avec différents profils de codage (tels que des résolutions, des débits binaires, des codecs ou des longueurs de segment variables) peuvent générer des NO_VARIANT_MATCH erreurs si le CTP ne prend pas en compte toutes les combinaisons de variantes présentes dans la bibliothèque. Si vous avez besoin d'un CTP unique pour prendre en charge plusieurs origines ou d'une VOD bibliothèque avec des profils de codage mixtes, il est possible de créer un CTP surdéfini qui englobe les variantes de toutes les origines cibles. Cependant, les CTP surdéfinis sont soumis à une limite stricte de 36 variantes vidéo, ce qui limite le nombre d'origines ou de combinaisons de variantes pouvant être prises en compte dans un seul profil.

Important

Vous devez synchroniser votre CTP chaque fois que vous modifiez les paramètres de votre encodeur ou que vous utilisez de nouvelles sources de VOD contenu. Le non-respect de cette consigne peut entraîner NO_VARIANT_MATCH des erreurs et des échecs de transcodage des annonces. Cela inclut des modifications telles que l'ajout ou la suppression de résolutions, l'ajustement des débits, la modification des codecs, la modification de la longueur des segments ou l'intégration de VOD contenu provenant de nouveaux fournisseurs avec des profils d'encodage différents.

Lorsque vous mettez à jour votre CTP, tenez compte du comportement suivant.

- La mise à jour d'un CTP existant applique les nouveaux paramètres uniquement aux annonces récemment transcodées. Les publicités précédemment transcodées selon les anciens paramètres CTP ne seront pas retranscodées.
- La création d'un nouveau CTP et son association à votre MediaTailor configuration vous inviteront MediaTailor à retranscoder toutes les annonces à l'aide du nouveau profil, garantissant ainsi une cohérence totale dans l'ensemble de votre inventaire publicitaire.

C'est pourquoi, lorsque les paramètres de votre encodeur changent ou que vous intégrez de nouvelles sources de contenu, il est recommandé de créer un nouveau CTP plutôt que de mettre à jour le CTP existant afin de garantir que toutes les publicités, y compris les publicités précédemment transcodées, reflètent les paramètres mis à jour.

Quand utiliser un profil de transcodage personnalisé

Utilisez un profil de transcodage personnalisé lorsque vos flux de travail nécessitent un ou plusieurs des éléments suivants.

- Une cible de normalisation audio personnalisée autre que la cible MediaTailor par défaut de -24 LKFS
- Non-integer longueurs de segment (par exemple, 1,92 ou 3,84)
- Configurations audio Dolby Vision
- Valeurs PID PMT, vidéo ou audio spécifiques requises par les systèmes ou lecteurs en aval
- Taux d'échantillonnage audio personnalisés (par exemple, 44,1 kHz)
- Correspondance du débit au-delà du seuil de 5 % pris en charge par le transcodage dynamique avec un indicateur de configuration
- Rapports hauteur/largeur en pixels (PAR) personnalisés, par exemple, 1:1 pour les pixels carrés ou 40:33 pour un écran large NTSC D1/DV
- Pistes audio ou sous-titrées nommées spécifiques

Commencer à utiliser des profils de transcodage personnalisés

L'accès à l'API de MediaTailor transcodage, qui est nécessaire pour enregistrer un CTP, est limité aux comptes autorisés. Pour commencer, contactez AWS Support en utilisant l'une des options suivantes.

- Demander l'accès à la liste des autorisations : pour créer et gérer vos propres CTP, contactez le support AWS pour demander que votre compte soit ajouté à la liste des autorisations. Une fois votre compte ajouté à la liste des autorisations, vous avez accès à l'API de transcodage pour enregistrer et gérer les profils dans toutes les AWS régions où vous disposez d'une configuration existante MediaTailor .
- Demander la création d'un CTP par AWS Support — Pour qu'AWS Support génère et enregistre un CTP en votre nom, contactez AWS Support avec une URL de point de terminaison autre que celle DRM-protected d'origine à laquelle MediaTailor vous pouvez accéder. AWS Support utilisera ce point de terminaison pour analyser votre configuration d'origine et générer un CTP approprié pour votre compte.

Transcodage dynamique et profils de transcodage personnalisés

Utilisez le tableau suivant pour déterminer l'approche adaptée à votre flux de travail.

Comparaison entre le transcodage dynamique et le profil de transcodage personnalisé

Considération	Transcodage dynamique	Profil de transcodage personnalisé
Effort de configuration	Aucune	Configuration de tâche AWS MediaConvert Elemental créée manuellement
Support du Trickplay (et) HLS DASH	Automatique	Doit être explicitement configuré dans le CTP
DASHManifestes compacts	Automatique	Doit être explicitement configuré dans le CTP
CMAFprise en charge des formats	Automatique	Doit être explicitement configuré dans le CTP
Gestion des modifications d'origine	Automatique	Mise à jour manuelle du CTP requise
Multi-region complexité	Aucune	Doit être enregistré par région
Normalisation audio	-24 LKFS (par défaut)	Entièrement configurable

Considération	Transcodage dynamique	Profil de transcodage personnalisé
Correspondance du débit	Jusqu'à 5 % d'arrondissement par <code>enableBitrateRounding</code> drapeau (contactez le support AWS pour activer)	Configuré explicitement dans le CTP pour correspondre aux débits d'origine
Non-integer longueurs des segments	2 secondes par défaut ; les longueurs de segment non entières ne sont pas prises en charge (contactez AWS Support <code>calculateSegmentLength</code> pour activer le calcul automatique de la longueur des segments entiers)	Vous devez configurer manuellement la longueur du segment pour qu'elle corresponde à l'origine
Valeurs PID personnalisées	Non pris en charge	Pris en charge
Système audio Dolby Vision	Non pris en charge	Pris en charge
Taux d'échantillonnage audio personnalisés	Non pris en charge	Pris en charge
Multi-origin soutien	Automatique	CTP Superset requis ; limité à 36 variantes vidéo
MediaTailor améliorations	Automatique	Mises à jour manuelles du CTP requises
Idéal pour	Correspondance de codecs standard, profils de débit communs, trickplay, compact DASH	Exigences de codage spécialisées non satisfaites par le transcodage dynamique

Publicités de prélecture

Utilisez la prélecture des AWS Elemental MediaTailor annonces pour les diffusions en direct afin de réduire la charge maximale sur les serveurs de décision publicitaire (ADS) et de réduire le temps de latence de diffusion des manifestes au début de chaque pause publicitaire. Lorsque vous définissez un calendrier de prélecture, suivez le MediaTailor calendrier pour récupérer les publicités de l'ADS et les préparer à l'insertion avant qu'elles ne soient nécessaires pour une pause publicitaire. Pendant les diffusions en direct, le préchargement peut contribuer à atténuer la baisse du taux de remplissage des publicités et les opportunités de monétisation manquées en raison des délais d'attente des demandes publicitaires et du transcodage ou d'autres retards sur le réseau.

Note

La prélecture des annonces ne fonctionne pas avec les méthodes d'insertion de publicités guidées par le serveur (SGAI), notamment les méthodes traditionnelles guidées par le serveur et les interstitiels HLS. Les méthodes SGAI ne nécessitent pas de prélecture, car les joueurs ne récupèrent que les publicités auxquelles ils vont jouer, et les manifestes peuvent être diffusés par des CDN sans MediaTailor recevoir de demandes de session individuelles.

Pour configurer le préchargement des publicités, vous devez créer un ou plusieurs programmes de prélecture dans votre configuration de diffusion. Un calendrier de prélecture indique MediaTailor comment et quand récupérer et préparer les publicités pour une prochaine pause publicitaire.

- Si un événement présente des disponibilités publicitaires selon un calendrier prévisible, utilisez un seul calendrier de prélecture. Chaque calendrier de prélecture définit un ensemble unique d'annonces MediaTailor à placer dans un seul et même catalogue publicitaire. Pour prévisualiser des annonces pour plusieurs mises à disposition lorsque vous utilisez des programmes de prélecture uniques, vous devez créer plusieurs programmes de prélecture (jusqu'à 24 heures avant la publication de l'annonce) qui correspondent à chaque disponibilité publicitaire.
- Si un événement présente des disponibilités publicitaires qui ne sont pas prévues selon un calendrier prévisible, utilisez un calendrier de prélecture récurrent. Un planning de prélecture récurrent crée automatiquement un calendrier et prévisualise les annonces avant chaque pause publicitaire d'un événement. Le programme de prélecture récurrent permet de récupérer les annonces pour chaque annonce publiée dans un délai défini (jusqu'à 24 heures avant la fin de l'événement). Vous n'avez pas besoin de créer un calendrier pour chaque publication publicitaire, mais vous perdez une partie du contrôle du temps qu'offre le préchargement unique.

Les rubriques suivantes fournissent des informations supplémentaires sur le préchargement des annonces.

Rubriques

- [Comment fonctionne le préchargement](#)
- [Création de plannings de prélecture](#)
- [TPS-based modelage du trafic](#)
- [Supprimer des programmes de prélecture](#)

Comment fonctionne le préchargement

Lorsque votre client envoie une demande de manifeste à MediaTailor, le service évalue tous les programmes de prélecture associés à la configuration de lecture. S'il MediaTailor ne trouve pas de calendrier de prélecture correspondant, le service revient à l'insertion normale des annonces et ne précharge pas les annonces.

S'il MediaTailor trouve un calendrier de prélecture correspondant, le service évalue le calendrier en fonction de deux éléments : l'extraction et la consommation. La configuration de chaque composant varie entre des programmes de prélecture uniques et des programmes de prélecture récurrents, comme décrit dans les sections suivantes.

Flux de planification de prélecture unique

Récupération

Cela définit la fenêtre de récupération, qui est la plage de temps pendant laquelle les publicités sont MediaTailor préluées depuis l'ADS. Assurez-vous de planifier cette fenêtre avant la pause publicitaire. Vous trouverez ci-dessous une vue d'ensemble de la manière dont MediaTailor un seul planning de prélecture est traité.

Pour savoir comment créer un calendrier de prélecture unique dans la console, consultez.

[Création de plannings de prélecture](#) Pour les instructions relatives à l'API, reportez-vous [PrefetchSchedules](#) à la référence de AWS Elemental MediaTailor l'API.

Au cours de la période de récupération spécifiée, MediaTailor envoie des demandes à l'ADS pour récupérer et préparer les publicités en vue de leur insertion ultérieure dans les sessions de diffusion.

- MediaTailor utilise éventuellement la mise en forme du trafic pour limiter le nombre de demandes adressées à l'ADS à la fois. Vous pouvez choisir entre deux approches :

Time-window mise en forme du trafic : MediaTailor répartit les demandes sur le nombre de secondes spécifié au lieu d'envoyer des demandes pour toutes les sessions en une seule fois. Cette répartition du trafic dispersé permet d'éviter que l'ADS ne soit submergé, ce qui entraîne des temps morts et de faibles taux de remplissage des publicités.

TPS-based mise en forme du trafic : MediaTailor limite les demandes en fonction des transactions par seconde (TPS) et des utilisateurs simultanés. Cette approche fournit une configuration plus intuitive basée sur les limites de capacité de votre ADS. Pour de plus amples informations, veuillez consulter [TPS-based modelage du trafic](#).

- Si vous configurez des variables dynamiques, MediaTailor incluez ces variables dans les demandes adressées à l'ADS. MediaTailor utilise ces variables pour faire correspondre les disponibilités publicitaires afin de prévisualiser les plannings pendant la fenêtre de consommation. Consultez la section Consommation suivante pour plus d'informations.

Exemple

Un événement en direct dure de 7 h 45 à 10 h, avec une pause publicitaire à 8 h 15. Vous configurez MediaTailor pour récupérer les publicités entre 7 h 45 et 8 h, avec une fenêtre de régulation du trafic de 60 secondes. Avec 500 000 utilisateurs simultanés, MediaTailor distribue les demandes ADS pour atteindre un taux moyen d'environ 8 333 transactions par seconde pendant 60 secondes ($500\,000 \text{ users} / 60 \text{ secondes} = 8\,333 \text{ demandes par seconde}$), au lieu d'envoyer toutes les demandes simultanément.

La configuration de récupération inclut la clé `scte.event` et la valeur 1234 de la variable dynamique. MediaTailor inclut cette variable dans les demandes adressées à l'ADS, qui peuvent ensuite être utilisées pour cibler des annonceurs spécifiques en fonction de l'ID d'événement 1234.

Consommation

Lorsqu'il MediaTailor rencontre des marqueurs de rupture de SCTE-35 publicité pendant la période de consommation, les publicités préluées sont placées dans un break publicitaire.

- Si vous n'avez pas défini de critères de correspondance, MediaTailor insère des annonces dès le premier intervalle de la fenêtre de consommation.
- Si vous avez défini une clé de variable dynamique pour utiliser les critères de correspondance, MediaTailor évalue ces critères par rapport aux variables dynamiques que vous avez définies

dans la fenêtre de récupération. Un break publicitaire ne peut être inséré dans une annonce préluée que si les critères de disponibilité sont remplis. MediaTailor insère des annonces dans la première pause qui répond aux critères.

Pour obtenir la liste des critères de disponibilité pris en charge, consultez la colonne Disponible pour la prélecture des annonces dans le tableau ci-dessous. [MediaTailor variables de session pour les requêtes ADS](#)

Exemple a continué

Vous définissez l'heure de début de la consommation à 8 h 15 et l'heure de fin à 8 h 17. Vous incluez `scte.event_id` la clé dans les critères de correspondance disponibles.

Pour chaque pause publicitaire entre 8 MediaTailor h 15 et 8 h 17, il évalue l'identifiant de l'SCTE événement pour chaque pause publicitaire. À chaque session de diffusion, MediaTailor insère les publicités préluées dans le premier break publicitaire dont l'identifiant d'événement est 1234 (tel que défini dans les variables dynamiques de récupération). Pour les pauses publicitaires qui ne contiennent pas le bon identifiant d'événement, MediaTailor effectue une insertion publicitaire standard.

Flux de planification de prélecture récurrent

Récupération

Cela définit la fenêtre de récupération récurrente, c'est-à-dire la plage de temps pendant laquelle les publicités sont MediaTailor préluées et insérées pour un événement en direct (jusqu'à 24 heures). Vous trouverez ci-dessous un aperçu de la manière dont MediaTailor les plannings de prélecture récurrents sont traités.

Pour savoir comment créer un calendrier de prélecture récurrent dans la console, consultez. [Création de plannings de prélecture](#) Pour les instructions relatives à l'API, reportez-vous [PrefetchSchedules](#) à la référence de AWS Elemental MediaTailor l'API.

Pendant la période de prélecture récurrente spécifiée, MediaTailor récupère et insère des publicités pour un événement en direct d'une durée maximale de 24 heures. Après chaque pause publicitaire dans la fenêtre, récupère MediaTailor automatiquement les annonces pour la pause publicitaire suivante.

- Si vous définissez le délai après la fin de la période d' MediaTailor utilisation, attendez le temps spécifié avant de récupérer le prochain ensemble d'annonces pour le prochain break publicitaire.
- MediaTailor utilise éventuellement la mise en forme du trafic pour limiter le nombre de demandes adressées à l'ADS à la fois. Vous pouvez choisir entre deux approches :

Time-window mise en forme du trafic : MediaTailor répartit les demandes sur le nombre de secondes spécifié au lieu d'envoyer des demandes pour toutes les sessions en une seule fois. Cette répartition du trafic dispersé permet d'éviter que l'ADS ne soit submergé, ce qui entraîne des temps morts et de faibles taux de remplissage des publicités.

TPS-based mise en forme du trafic : MediaTailor limite les demandes en fonction des transactions par seconde (TPS) et des utilisateurs simultanés. Cette approche fournit une configuration plus intuitive basée sur les limites de capacité de votre ADS. Pour de plus amples informations, veuillez consulter [TPS-based modelage du trafic](#).

- Si vous configurez des variables dynamiques, MediaTailor incluez ces variables dans les demandes adressées à l'ADS. MediaTailor utilise ces variables pour faire correspondre les disponibilités publicitaires afin de prévisualiser les plannings pendant la fenêtre de consommation. Consultez la section Consommation suivante pour plus d'informations.

Exemple

Un événement en direct dure de 19 h à 20 h 45, avec quatre pauses publicitaires pendant cette période. Les pauses publicitaires ne se déroulent pas selon un calendrier prévisible. Vous configurez le prefetch récurrent de 19 h 00 à 20 h 45, avec un délai de 10 minutes et une fenêtre de régulation du trafic de 60 secondes. Après chaque utilisation, MediaTailor récupère les publicités pour la prochaine pause publicitaire. Dix minutes après la fin de l'utilisation, MediaTailor commence à envoyer des demandes de récupération à l'ADS. Avec une fenêtre de régulation du trafic de 60 secondes et 500 000 utilisateurs simultanés, MediaTailor distribue les demandes ADS pour atteindre un taux moyen d'environ 8 333 transactions par seconde pendant 60 secondes ($500\,000 \text{ secondes users} / 60 = 8\,333 \text{ demandes par seconde}$), au lieu d'envoyer toutes les demandes simultanément.

La configuration de récupération inclut la clé `scte.event` et la valeur 1234 de la variable dynamique. MediaTailor inclut cette variable dans les demandes adressées à l'ADS, qui peuvent ensuite être utilisées pour cibler des annonceurs spécifiques en fonction de l'ID d'événement 1234.

Consommation

Lorsqu'il MediaTailor rencontre des marqueurs de rupture de SCTE-35 publicité, il place les annonces préluées dans un break publicitaire.

- Si vous définissez la date d'expiration des annonces récupérées, les annonces préluées peuvent être insérées jusqu'à la date d'expiration spécifiée.
- Si vous n'avez pas défini de critères de correspondance, MediaTailor insère des annonces dès le premier intervalle de la fenêtre de consommation.
- Si vous avez défini une clé de variable dynamique pour utiliser les critères de correspondance, MediaTailor évalue ces critères par rapport aux variables dynamiques que vous avez définies dans la fenêtre de récupération. Un break publicitaire ne peut être inséré dans une annonce préluée que si les critères de disponibilité sont remplis. MediaTailor insère des annonces dans la première pause qui répond aux critères.

Pour obtenir la liste des critères de disponibilité pris en charge, consultez la colonne Disponible pour la prélecture des annonces dans le tableau ci-dessous. [MediaTailor variables de session pour les requêtes ADS](#)

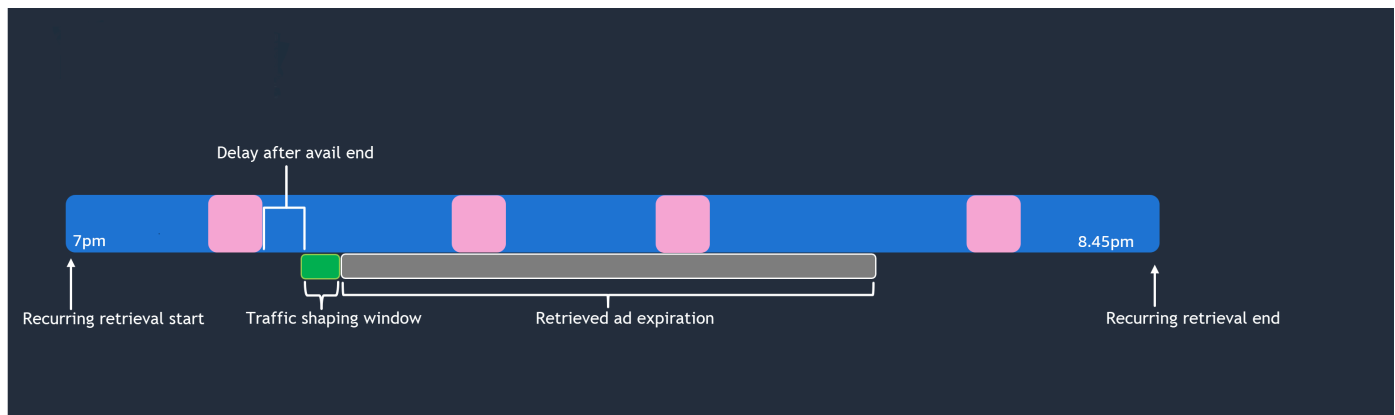
Exemple a continué

Dans la consommation, vous incluez `scte.event_id` la clé dans les critères de correspondance disponibles.

Pour chaque pause publicitaire MediaTailor rencontrée, il évalue l'identifiant de l'SCTE événement correspondant à chaque interruption publicitaire. À chaque session de diffusion, MediaTailor insère les publicités préluées dans chaque pause publicitaire dont l'identifiant d'événement est 1234 (tel que défini dans les variables dynamiques de récupération). Pour les pauses publicitaires qui ne contiennent pas le bon identifiant d'événement, MediaTailor effectue une insertion publicitaire standard.

Vous avez défini l'expiration des annonces à 2 700 secondes afin que les annonces récupérées puissent être insérées pendant 45 minutes.

Le graphique suivant illustre cet exemple, avec les petits carrés représentant les pauses publicitaires. Les paramètres du calendrier de prélecture récurrent sont illustrés le long de la chronologie de l'événement.



Comprendre les coûts de prélecture

Les demandes de récupération de publicités sont gratuites. Toutefois, pour la récupération des publicités préluées, vous serez facturé au tarif de transcodage standard pour les publicités préluées qui transcodent. MediaTailor Pour la consommation de publicités préluées, vous serez facturée au tarif standard pour l'insertion d'annonces préluées placées dans des pauses publicitaires. MediaTailor Pour plus d'informations sur les coûts de transcodage et d'insertion de publicités, consultez la section [AWS Elemental MediaTailor Tarification](#).

Création de plannings de prélecture

La procédure suivante explique comment créer un planning de prélecture à l'aide de la MediaTailor console. Pour plus d'informations sur la création et la gestion des plannings de prélecture par programmation à l'aide de l' API MediaTailor, consultez la référence de [PrefetchSchedules](#) l'API AWS Elemental MediaTailor.

Note

Lorsque vous configurez des programmes de prélecture dans MediaTailor, il est important de comprendre comment les différents types de variables sont gérés.

Utiliser les critères de correspondance

Si vous souhaitez utiliser des critères de correspondance de disponibilité dans un calendrier, assurez-vous d'abord de configurer le modèle d'URL ADS de votre configuration de lecture avec des [variables de session dynamiques](#), sinon les critères de correspondance de disponibilité n'auront aucun effet. Pour plus d'informations sur l'utilisation de variables de session dynamiques, consultez [Étape 3 : Configuration de](#)

[l'URL de la demande ADS et des paramètres de requête](#) la rubrique Commencer à insérer des MediaTailor annonces.

Variables relatives aux joueurs dans les plannings de prélecture

Lorsque vous créez un planning de prélecture, ne définissez pas les variables du joueur en tant que variables dynamiques dans votre configuration de prélecture. Transmettez plutôt les variables du joueur comme vous le feriez normalement au début de la session. MediaTailor inclut automatiquement ces variables dans les demandes publicitaires de prélecture si les variables sont mappées dans l'URL du modèle ADS.

Pour créer un nouveau calendrier de prélecture à l'aide de la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, sélectionnez Configurations. Sélectionnez la configuration de lecture pour laquelle vous souhaitez créer un calendrier de prélecture.
3. Dans l'onglet Programmes de prélecture, choisissez Ajouter un calendrier de prélecture.
4. Dans le volet des détails du planning Prefetch, procédez comme suit :
 - Dans Nom, entrez un identifiant pour votre planning de prélecture, tel que my-prefetch-schedule.
 - Pour Stream ID, entrez éventuellement un identifiant unique. Si votre source contient plusieurs flux de lecture, vous pouvez utiliser cet identifiant pour demander de MediaTailor placer des publicités dans un flux spécifique. Par exemple, si votre configuration de diffusion comprend une diffusion sportive et une diffusion d'émission de télévision, vous pouvez utiliser l'ID de diffusion pour créer des programmes de prélecture afin d'insérer des publicités ciblées pour la diffusion sportive. Vous transmettez la valeur de l'ID de flux MediaTailor lors de l'initialisation de la session ou de la demande de manifeste de votre client. Pour plus d'informations, consultez l'exemple suivant.
 - Pour le suivi côté serveur, incluez le paramètre et la valeur de la `?aws.streamId` requête dans la GET HTTP demande envoyée par votre client à votre MediaTailor point de terminaison. Pour des informations générales sur le suivi côté serveur, voir. [MediaTailor suivi des publicités et rapports côté serveur](#) Une demande de manifeste adressée à un point de terminaison HLS qui inclut un identifiant de flux se présente comme suit, où se *myStreamId* trouve le nom de votre identifiant de flux :

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?
aws.streamId=myStreamId
```

- Pour le suivi côté client, incluez la `streamId` clé et la valeur dans le corps de la demande d'initialisation de POST HTTP session de votre client envoyée au point de terminaison. `MediaTailor/v1/session` Pour des informations générales sur le suivi côté client, voir. [Client-side suivi des publicités](#) Une demande d'initialisation de session qui inclut un identifiant de flux se présente comme suit, où se `myStreamId` trouve le nom de votre identifiant de flux :

```
POST <mediatailorURL>/v1/session/<hashed-account-id>/<origin-id>/<asset-id>
{
  'streamId': 'myStreamId',
  'reportingMode': 'client'
}
```

5. Pour le type Prefetch, faites votre sélection et choisissez la section correspondante pour obtenir de l'aide sur les champs supplémentaires :
 - Choisissez Single si vous créez un calendrier de prélecture pour une pause publicitaire dans le cadre d'un événement.
 - Choisissez Récurrent si vous créez un calendrier qui prévisualise automatiquement les annonces avant chaque interruption publicitaire d'un événement.

Programme de prélecture unique

Pour créer un calendrier prévisualisant les annonces avant qu'une annonce ne soit publiée dans le cadre d'un événement.

1. Dans le volet Extraction, spécifiez les paramètres de récupération que vous souhaitez utiliser. Ces paramètres déterminent à quel moment les publicités sont MediaTailor extraites de l'ADS. Ils déterminent également les variables de session dynamiques à inclure dans la demande adressée à l'ADS, le cas échéant.
 - Dans le champ Heure de début, entrez l'heure à laquelle les extractions par prélecture MediaTailor peuvent commencer pour cette pause publicitaire. MediaTailor tentera de prévisualiser les publicités pour les demandes manifestes faites par votre client à cette date ou après cette date. La valeur par défaut est l'heure actuelle. Si vous ne spécifiez aucune valeur, le service lance la récupération par prélecture dès que possible.

- Dans Heure de fin, entrez l'heure à laquelle vous MediaTailor souhaitez arrêter de prévisualiser les publicités pour cette pause publicitaire. MediaTailor tentera de prévisualiser les publicités pour les demandes manifestes qui se produisent à cette heure ou avant. La fenêtre de récupération peut se chevaucher avec la fenêtre de consommation.
- Configurez éventuellement la mise en forme du trafic pour limiter le nombre de demandes adressées à l'ADS à la fois. Choisissez l'une des approches suivantes :

Time-window approche : pour la durée de la fenêtre de façonnage du trafic, entrez le nombre de secondes pendant lesquelles les demandes MediaTailor doivent être distribuées à l'ADS. Pour plus d'informations, voir Explication de la [récupération du calendrier par prélecture unique](#).

TPS-based approche : configurez les utilisateurs simultanés Peak TPS et Peak pour limiter les demandes en fonction des transactions par seconde et des utilisateurs simultanés. Pour de plus amples informations, veuillez consulter [TPS-based modelage du trafic](#).

- Dans la section [Variables dynamiques](#), entrez jusqu'à 100 variables de session dynamiques. MediaTailor utilise ces variables à des fins de substitution dans les demandes de prélecture qu'il envoie à l'ADS. Si vous ne saisissez aucune variable de session dynamique MediaTailor, essayez d'interpoler les valeurs des variables dynamiques contenues dans votre URL [ADS](#).
- Sélectionnez Ajouter une variable dynamique.
- Pour Clé, entrez une clé de variable de session dynamique, telle que `scte.event_id`. Vous pouvez utiliser n'importe quelle variable dynamique que MediaTailor prend en charge. Pour plus d'informations sur les variables de session dynamiques, consultez [MediaTailor variables de session pour les requêtes ADS](#).
- Pour Valeur, entrez une valeur de variable dynamique, telle que `my-event`.
- Pour ajouter une autre variable dynamique, choisissez Sélectionner Ajouter une variable dynamique.

2. Dans le volet Consommation, spécifiez les paramètres que vous souhaitez utiliser pour la fenêtre de consommation. Ces paramètres déterminent à quel moment MediaTailor les annonces sont placées dans le break publicitaire. Ils déterminent également les critères de correspondance de disponibilité que vous souhaitez utiliser.

- Pour Heure de début, entrez l'heure à laquelle vous souhaitez commencer MediaTailor à placer des annonces préluées dans le break publicitaire. La valeur par défaut est l'heure actuelle. Si vous ne spécifiez pas d'heure, le service démarre la consommation de prélecture dès que possible.

- Pour Heure de fin, entrez l'heure à laquelle vous MediaTailor souhaitez arrêter de placer les publicités préluées dans le break publicitaire. MediaTailor tentera de prévisualiser les publicités pour les demandes manifestes de votre client qui se produisent à cette heure ou avant. L'heure de fin doit être postérieure à l'heure de début, et dans moins d'un jour. La fenêtre de consommation peut se chevaucher avec la fenêtre de récupération.
- Dans la section [Critères de disponibilité](#), sélectionnez Ajouter des critères de disponibilité et ajoutez jusqu'à cinq critères de correspondance de disponibilité à votre calendrier. Ensuite, sous Clé variable dynamique, ajoutez une clé variable dynamique, telle que `scte.event_id`. MediaTailor placera les annonces préluées dans le break publicitaire uniquement si elles répondent aux critères définis par les valeurs des variables dynamiques que votre client transmet ou MediaTailor déduites à MediaTailor partir d'informations telles que les données de session. Si un break publicitaire ne répond pas aux critères de correspondance spécifiés, MediaTailor ignore le préchargement correspondant à cet intervalle. Pour plus d'informations, consultez l'explication de la [consommation du programme de prélecture unique](#).

 Important

Vous pouvez spécifier un maximum de cinq critères de correspondance de disponibilité par calendrier de prélecture. Si vous spécifiez plusieurs critères, MediaTailor combinez-les à l'aide d'un ET logique. Pour modéliser des combinaisons logiques OR, créez plusieurs programmes de prélecture.

Utilisez uniquement des variables qui changent par pause publicitaire (telles que `scte.event_id`, `scte.segmentation_upid`, ou `avail.index`) pour répondre aux critères de correspondance. N'utilisez pas de variables de joueur (telles que `player_params.*`) pour utiliser des critères de correspondance, car les variables relatives aux joueurs restent constantes pendant toute la session et ne font pas de différence entre les pauses publicitaires. MediaTailor inclut automatiquement les variables relatives aux joueurs dans les demandes publicitaires de prélecture si les variables sont mappées dans l'URL du modèle ADS.

3. Sélectionnez Ajouter des critères de disponibilité.

Les programmes Prefetch expirent automatiquement après la fin de la fenêtre de consommation. À des fins de diagnostic, ils restent visibles pendant au moins 7 jours, après quoi ils MediaTailor sont automatiquement supprimés. Vous pouvez également supprimer manuellement un planning de prélecture à tout moment. Pour plus d'informations sur la suppression manuelle d'un planning

de prélecture, consultez la section suivante [the section called “Supprimer des programmes de prélecture”](#).

Déterminer la fréquence à laquelle votre client doit appeler l' `CreatePrefetchSchedule` API

Votre client peut appeler l' [CreatePrefetchSchedule](#) API par programmation une fois par jour pour configurer la récupération et la consommation si vous savez exactement quand se produiront les interruptions publicitaires. Votre client peut également appeler l'API plusieurs fois au cours de la journée pour définir la récupération et la consommation. Lorsque vous choisissez la fréquence des appels d'API, tenez compte du [nombre maximum de programmes de prélecture actifs](#) et de la probabilité que votre calendrier de pause publicitaire change une fois que vous aurez créé votre ou vos programmes de prélecture. S'il est probable que le calendrier des pauses publicitaires change une fois que vous avez créé vos programmes de prélecture, vous pouvez appeler l'API plus fréquemment.

Programme de prélecture récurrent

Pour créer un calendrier prévisualisant les annonces avant que chaque annonce ne soit publiée dans le cadre d'un événement.

1. Dans le volet Récupération récurrente, spécifiez les paramètres de récupération que vous souhaitez utiliser. Ces paramètres déterminent à quel moment les publicités sont MediaTailor extraites de l'ADS. Ils déterminent également les variables de session dynamiques à inclure dans la demande adressée à l'ADS, le cas échéant.
 - Pour la fenêtre de prélecture récurrente, entrez l'heure à laquelle les extractions de prélecture MediaTailor peuvent commencer pour cette pause publicitaire. MediaTailor tentera de prévisualiser les publicités pour les demandes manifestes faites par votre client à cette date ou après cette date. La valeur par défaut est l'heure actuelle. Si vous ne spécifiez aucune valeur, le service lance la récupération par prélecture dès que possible.
 - Dans le champ Délai après la fin d'utilisation, entrez le nombre de secondes qui MediaTailor doivent s'écouler après la fin d'une utilisation avant de prévisualiser les publicités pour la prochaine utilisation. Si vous ne spécifiez aucune valeur, la valeur MediaTailor par défaut est « aucun délai ».
 - Configurez éventuellement la mise en forme du trafic pour limiter le nombre de demandes adressées à l'ADS à la fois. Choisissez l'une des approches suivantes :

Time-window approche : pour la durée de la fenêtre de façonnage du trafic, entrez le nombre de secondes pendant lesquelles les demandes MediaTailor doivent être distribuées à l'ADS.

Pour plus d'informations, voir Explication de la [récupération du calendrier de prélecture récurrent](#)

TPS-based approche : configurez les utilisateurs simultanés Peak TPS et Peak pour limiter les demandes en fonction des transactions par seconde et des utilisateurs simultanés. Pour de plus amples informations, veuillez consulter [TPS-based modelage du trafic](#).

- Dans la section [Variables dynamiques](#), entrez jusqu'à 100 variables de session dynamiques. MediaTailor utilise ces variables à des fins de substitution dans les demandes de prélecture qu'il envoie à l'ADS. Si vous ne saisissez aucune variable de session dynamique MediaTailor, essayez d'interpoler les valeurs des variables dynamiques contenues dans votre URL [ADS](#).
 - Sélectionnez Ajouter une variable dynamique.
 - Pour Clé, entrez une clé de variable de session dynamique, telle que `scte.event_id`. Vous pouvez utiliser n'importe quelle variable dynamique que MediaTailor prend en charge. Pour plus d'informations sur les variables de session dynamiques, consultez [MediaTailor variables de session pour les requêtes ADS](#).
 - Pour Valeur, entrez une valeur de variable dynamique, telle que `my-event`.
 - Pour ajouter une autre variable dynamique, choisissez Sélectionner Ajouter une variable dynamique.
2. Dans le volet Consommation, spécifiez les paramètres que vous souhaitez utiliser pour la fenêtre de consommation. Ces paramètres déterminent à quel moment MediaTailor les annonces sont placées dans le break publicitaire. Ils déterminent également les critères de correspondance de disponibilité que vous souhaitez utiliser.
- Pour Expiration des annonces récupérées, indiquez combien de temps après la récupération les annonces peuvent être insérées.
 - Dans la section [Critères de disponibilité](#), sélectionnez Ajouter des critères de disponibilité et ajoutez jusqu'à cinq critères de correspondance de disponibilité à votre calendrier. Ensuite, sous Clé variable dynamique, ajoutez une clé variable dynamique, telle que `scte.event_id`. MediaTailor placera les annonces préluées dans le break publicitaire uniquement si elles répondent aux critères définis par les valeurs des variables dynamiques que votre client transmet ou MediaTailor déduites à MediaTailor partir d'informations telles que les données de session. Si un break publicitaire ne répond pas aux critères de correspondance spécifiés, MediaTailor ignore le préchargement correspondant à cet intervalle. Pour plus d'informations, consultez l'explication de la [consommation récurrente du programme de prélecture](#).

⚠ Important

Vous pouvez spécifier un maximum de cinq critères de correspondance de disponibilité par calendrier de prélecture. Si vous spécifiez plusieurs critères, MediaTailor combinez-les à l'aide d'un ET logique. Pour modéliser des combinaisons logiques OR, créez plusieurs programmes de prélecture.

Utilisez uniquement des variables qui changent par pause publicitaire (telles que `scte.event_id`, `scte.segmentation_upid`, ou `avail.index`) pour répondre aux critères de correspondance. N'utilisez pas de variables de joueur (telles que `player_params.*`) pour utiliser des critères de correspondance, car les variables relatives aux joueurs restent constantes pendant toute la session et ne font pas de différence entre les pauses publicitaires. MediaTailor inclut automatiquement les variables relatives aux joueurs dans les demandes publicitaires de prélecture si les variables sont mappées dans l'URL du modèle ADS.

3. Sélectionnez Ajouter des critères de disponibilité.

Les programmes Prefetch expirent automatiquement après la fin de la fenêtre de consommation. À des fins de diagnostic, ils restent visibles pendant au moins 7 jours, après quoi ils MediaTailor sont automatiquement supprimés. Vous pouvez également supprimer manuellement un planning de prélecture à tout moment. Pour plus d'informations sur la suppression manuelle d'un planning de prélecture, consultez la section suivante [the section called “Supprimer des programmes de prélecture”](#).

TPS-based modelage du trafic

AWS Elemental MediaTailor propose deux approches facultatives de mise en forme du trafic pour limiter le nombre de demandes adressées à l'ADS à la fois. TPS-based la mise en forme du trafic offre une alternative à la mise en forme du trafic basée sur des fenêtres temporelles pour les horaires de prélecture. Cette approche fournit une configuration plus intuitive en vous permettant de spécifier la capacité de votre serveur de décision publicitaire (ADS) en termes de transactions par seconde (TPS) et d'utilisateurs simultanés attendus, plutôt que de calculs de temps.

Comment fonctionne TPS-based le façonnage du trafic

Au lieu de spécifier les durées des fenêtres de récupération, vous fournissez les paramètres suivants :

TPS de pointe

Le nombre maximum de demandes par seconde que votre ADS peut traiter. Ce paramètre n'a aucune valeur par défaut.

Nombre maximal d'utilisateurs simultanés

Le nombre maximal attendu de spectateurs simultanés pour votre contenu. Ce paramètre n'a aucune valeur par défaut.

MediaTailor distribue automatiquement les demandes de prélecture dans le temps afin de respecter la limite TPS spécifiée, quel que soit le nombre de sessions simultanées.

Exemple TPS-based exemple de configuration

Votre ADS peut gérer 500 TPS et vous vous attendez à 100 000 spectateurs simultanés aux heures de pointe. Vous configurez :

- TPS de pointe : 500
- Nombre maximal d'utilisateurs simultanés : 100 000

MediaTailor distribue automatiquement les demandes de prélecture dans le temps afin de respecter la limite TPS spécifiée, quel que soit le nombre de sessions simultanées.

Supprimer des programmes de prélecture

La procédure suivante explique comment supprimer un planning de prélecture à l'aide de la MediaTailor console. Pour plus d'informations sur la suppression des plannings de prélecture par programmation à l'aide de l'API MediaTailor, consultez la référence de [DeletePrefetchSchedule](#) l'API AWS Elemental MediaTailor

Note

La suppression ne s'effectue pas en temps réel. Il se peut que vous rencontriez un certain délai lors de la MediaTailor suppression du ou des programmes de prélecture, période pendant laquelle la récupération et la consommation du préchargement continueront de s'exécuter en arrière-plan.

Pour supprimer un calendrier de prélecture à l'aide de la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, sélectionnez Configurations. Sélectionnez la configuration de lecture qui contient le ou les programmes de prélecture que vous souhaitez supprimer.
3. Dans l'onglet Programmes de prélecture, sélectionnez le calendrier de prélecture que vous souhaitez supprimer. Ensuite, choisissez Supprimer.

Utilisation de publicités préconditionnées avec AWS Elemental MediaTailor

Dans le cadre d'un [flux de travail d'insertion d'annonces classique](#), les annonces sont transcodées de MediaTailor manière dynamique pour qu'elles correspondent au flux de contenu, les enregistre et les intègre au flux en direct. Comme ce processus ne se produit qu'après MediaTailor réception de l'annonce dans une réponse VAST du serveur de décision publicitaire (ADS), il y a un délai avant que l'annonce soit disponible pour l'assemblage. Si une latence supplémentaire est introduite dans le flux de travail d'assemblage des annonces (soit en raison du délai d'attente de l'ADS, soit en raison d'un autre problème lié au contenu ou au réseau), MediaTailor cela pourrait partiellement combler le déficit ou complètement manquer la pause publicitaire.

Pour réduire le temps nécessaire à l'intégration des annonces dans votre contenu, vous pouvez utiliser des publicités préconditionnées. Une annonce préconditionnée est une annonce que vous transcodez avant de l'utiliser pour l'insertion d'annonces MediaTailor. Au lieu de fournir URLs des publicités non conditionnées à votre ADS, vous fournissez URLs des publicités préconditionnées. Dans sa réponse VAST à la MediaTailor demande, l'ADS inclut des liens directs vers les publicités préconditionnées. En supprimant la partie transcodage de l'assemblage des annonces, il MediaTailor suffit d'enregistrer l'annonce et de l'intégrer au flux de contenu. Le processus d'assemblage des publicités préconditionnées réduit le délai entre le moment où une annonce MediaTailor est signalée par le biais de la réponse VAST et le moment où l'annonce est intégrée au contenu.

Vous pouvez également utiliser le préchargement des annonces, qui consiste MediaTailor à configurer le processus d'assemblage des annonces à une heure planifiée avant que la pause publicitaire ne soit nécessaire. Pour plus d'informations sur le préchargement des annonces, consultez [Publicités de prélecture](#)

Exigences relatives aux annonces préconditionnées

Les exigences suivantes doivent être prises en compte lors de la configuration d'un flux de travail d'assemblage d'annonces avec des publicités préconditionnées.

Exigences relatives à **MediaFiles**

La réponse VAST à laquelle le serveur publicitaire envoie MediaTailor doit inclure MediaFiles les éléments suivants :

L'annonce (Creative) doit comporter des variantes conformes aux variantes de débit du flux de contenu. Il est de votre responsabilité de vous assurer que la réponse VAST utilise les variantes d'annonces appropriées pour correspondre aux modèles de manifestes.

Bien que l'utilisation de publicités préconditionnées puisse contribuer à rendre l'insertion des annonces plus efficace, MediaTailor elle n'est pas en mesure de gérer le processus de transcodage pour garantir que les fichiers multimédias des publicités sont compatibles avec les spécifications du manifeste de contenu. Si l'annonce ne correspond pas au flux de contenu, elle risque de ne pas MediaTailor être insérée ou la non-concordance peut provoquer une erreur sur l'appareil de lecture.

En outre, pour être intégré au flux de contenu sans MediaTailor transcodage, MediaFile il doit répondre aux exigences suivantes :

- Il doit être accessible sur l'Internet public pour MediaTailor pouvoir le télécharger.
- Il doit utiliser la diffusion en continu, désignée comme `delivery="streaming"` dans la réponse VAST.
- Il doit s'agir d'un fichier `.m3u8` (pour HLS) ou `.mpd` (pour DASH).

Exemple Réponse VASTE

À partir de l'exemple de réponse VAST suivant, MediaTailor insère le MediaFile texte suivant URLs :

- Pour un flux HLS, MediaTailor utilise `https://example-ad-origin.amazonaws.com/ad1/index_low.m3u8`. C'est le premier à proposer MediaFile une diffusion en streaming et une extension de fichier prise en charge (`.m3u8`).
- Pour un flux DASH, MediaTailor utilise `https://example-ad-origin.amazonaws.com/ad1/index.mpd`. C'est le premier à proposer MediaFile une diffusion en streaming et une extension de fichier prise en charge (`.mpd`).

```

<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="3.0">
  <Ad id="ad1">
    <InLine>
      <AdSystem>ExampleAdSystem</AdSystem>
      <AdTitle>ad1</AdTitle>
      <Impression><![CDATA[https://example-impression.amazonaws.com]]></
Impression>
      <AdServingId>de8e0d33-9c72-4d77-bb3a-f7e566ffc605</AdServingId>
      <Creatives>
        <Creative id="creativeId1" sequence="1">
          <Linear skipoffset="00:00:05">
            <Duration>00:00:30</Duration>
            <MediaFiles>
              <MediaFile delivery="progressive" width="1280" height="720"
type="video/mp4" bitrate="533" scalable="true" maintainAspectRatio="true"><![
CDATA[https://example-ad-origin.amazonaws.com/ad1/ad1.mp4]]></MediaFile>
              <MediaFile delivery="streaming" width="1280"
height="720" type="application/dash+xml" bitrate="533" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index.mpd]]></MediaFile>
              <MediaFile delivery="streaming" width="640"
height="360" type="application/x-mpegURL" bitrate="262" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_low.m3u8]]></MediaFile>
              <MediaFile delivery="streaming" width="2560"
height="1440" type="application/x-mpegURL" bitrate="1066" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_high.m3u8]]></MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
    </InLine>
  </Ad>
</VAST>

```

Exigences relatives aux manifestes publicitaires

Pour utiliser des publicités préconditionnées, les manifestes publicitaires de vos parents et enfants doivent répondre aux exigences suivantes :

- Le manifeste lié dans la Creative section de la réponse VAST doit être le manifeste publicitaire parent.
- Les URLs manifestes publicitaires destinés aux enfants doivent être des chemins relatifs.
- Les manifestes publicitaires pour enfants doivent se trouver dans le même répertoire que la playlist multivariante parente, au même niveau. Les manifestes enfants ne peuvent pas se trouver dans un sous-répertoire ou à un autre emplacement.

Exemple liste de lecture multivariante parent prise en charge

La liste de lecture multivariante suivante contient des listes de lecture multimédia relatives URLs aux annonces pour enfants. Les playlists enfants se trouvent également dans le même répertoire que la playlist multivariante parente.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
index_3.m3u8
```

Exemple liste de lecture multivariante parent non prise en charge : sous-répertoires

La liste de lecture multivariante d'annonces parent suivante contient des listes de lecture pour enfants qui se trouvent dans des sous-répertoires relatifs à la liste de lecture multivariante parent. Il ne s'agit pas d'une playlist compatible avec les publicités préconditionnées.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
child/index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
child/index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
child/index_3.m3u8
```

Exemple liste de lecture multivariante parent non prise en charge : absolue URLs

La liste de lecture multivariante suivante contient des listes de lecture pour enfants avec valeur absolue URLs. Il ne s'agit pas d'une playlist compatible avec les publicités préconditionnées.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_3.m3u8
```

Flux de travail publicitaire préconditionné

Vous trouverez ci-dessous une description de base du fonctionnement des publicités préconditionnées dans un flux de travail d'assemblage publicitaire avec MediaTailor. La première partie du flux de travail concerne les actions que vous devez effectuer pour vous préparer à utiliser des publicités préconditionnées. La deuxième partie décrit le MediaTailor traitement des publicités.

Partie 1 : Configuration des publicités préconditionnées

Procédez comme suit pour configurer un flux de travail qui utilise des publicités préconditionnées dans MediaTailor.

1. Utilisez un service de transcodeur, par exemple AWS Elemental MediaConvert, pour conditionner vos créations en variantes compatibles avec les différents débits, résolutions et codecs de vos modèles de manifestes.
2. Fournissez URLs les fichiers multimédias pré-transcodés à votre ADS, à utiliser dans les réponses VAST.
3. [Créez votre configuration de lecture](#) dans MediaTailor. Pour utiliser des publicités préconditionnées, sélectionnez Aucun pour le paramètre de conditionnement des fichiers multimédia en streaming dans la configuration.
4. Continuez à configurer la diffusion de votre contenu comme vous le feriez normalement.

Partie 2 : traitement des MediaTailor publicités

MediaTailor et l'assemblage se termine comme décrit dans. [Comment fonctionne MediaTailor l'insertion d'annonces](#) Lorsqu'il MediaTailor reçoit une réponse VAST de l'ADS, il utilise la logique suivante pour déterminer les actions à entreprendre pour les publicités. Cette logique est dictée par le paramètre de conditionnement des fichiers multimédia en streaming de la configuration de lecture.

- Lorsque le conditionnement des fichiers multimédia en streaming est défini sur Transcode, MediaTailor transcode les fichiers multimédia lors de leur progressive livraison et les assemble dans le manifeste. S'il n'y a pas assez de publicités progressive contenant des fichiers multimédia pour les diffuser, MediaTailor transcodez et utilisez ces fichiers pour les streaming diffuser.
- Lorsque le conditionnement des fichiers multimédia en streaming est défini sur Aucun, MediaTailor les publicités contenant des fichiers multimédia de streaming diffusion sont insérées dans le manifeste sans les transcoder. S'il n'y a pas assez de publicités streaming contenant des fichiers multimédia pour les diffuser, MediaTailor transcodez et utilisez ces fichiers pour les progressive diffuser.

Utilisation des fonctions de MediaTailor monétisation

Avec les fonctions de MediaTailor monétisation (Functions), vous pouvez personnaliser la façon dont vous gérez AWS Elemental MediaTailor les données de session et créez des demandes publicitaires lors de l'insertion des annonces. Les fonctions vous permettent d'appeler des API externes, de transformer les données à l'aide d'expressions et de modifier les paramètres de demande du serveur de décision publicitaire (ADS) ou les paramètres des joueurs. Vous n'avez pas besoin de déployer ou de gérer une infrastructure personnalisée.

Les fonctions utilisent JSonata, un langage de requête et de transformation léger pour les données JSON, pour évaluer les expressions. Vous écrivez des expressions JSonata pour lire les données de session, transformer les valeurs et définir la sortie. Pour obtenir la liste complète des fonctions JSonata prises en charge, consultez. [Référence d'expression JSonata](#)

Utilisez les fonctions lorsque vous devez :

- Enrichissez les demandes publicitaires avec les données des spectateurs. Appelez un service d'identité au début de la session et incluez l'identité résolue dans chaque demande ADS pour un ciblage publicitaire personnalisé.
- Personnalisez les paramètres de demande ADS. Définissez dynamiquement l'URL, les en-têtes ou le corps de l'ADS en fonction des données de session, SCTE-35 des signaux ou des réponses d'API externes.
- Effectuez A/B des tests sur les serveurs publicitaires. Répartissez le trafic entre les différents points de terminaison ADS en fonction des attributs de session ou d'une attribution aléatoire.
- Construisez des pipelines d'enrichissement en plusieurs étapes. Associez plusieurs fonctions pour récupérer les données d'une API, les transformer et transmettre les résultats à l'étape suivante.

Comment les fonctions s'intègrent dans le MediaTailor flux de travail

Lorsqu'un internaute lance une session de diffusion, il MediaTailor évalue vos fonctions à des moments spécifiques du flux d'insertion publicitaire, appelés « Lifecycle Hooks ».

1. Vous définissez une fonction et vous l'associez à une configuration de lecture par le biais d'un mappage de fonctions.
2. MediaTailor exécute la fonction au niveau du hook de cycle de vie désigné.
3. La fonction lit les données de session, appelle éventuellement une API externe et écrit les résultats qui sont MediaTailor utilisés lors du traitement de la lecture.

Tip

Si vous débutez dans le domaine des fonctions, commencez par le [Guide de démarrage rapide](#), puis lisez [Hooks de cycle de vie](#). Pour la syntaxe des expressions et des exemples complets, reportez-vous [Référence d'expression JSonata](#) aux sections et [Exemples de fonctions](#).

Rubriques

- [Guide de démarrage rapide des fonctions](#)
- [Fonctions, cycle de vie, crochets](#)
- [Types de fonctions et composition](#)
- [Création et gestion de fonctions](#)
- [Référence d'expression JSonata pour les fonctions](#)
- [Exemples de fonctions](#)
- [Fonctions de dépannage et de surveillance](#)
- [Limites des fonctions](#)

Guide de démarrage rapide des fonctions

Ce guide vous explique comment créer votre première fonction, l'associer à une configuration de lecture et vérifier qu'elle s'est exécutée. À la fin, vous disposerez d'une fonction fonctionnelle qui classifie le type d'appareil de chaque spectateur (cvt,mobile, oudesktop) et le stocke en tant que paramètre du joueur disponible dans chaque requête ADS.

Conditions préalables

Avant de commencer, assurez-vous que vous disposez d'une configuration MediaTailor de lecture existante. Si vous n'en avez pas, veuillez consulter [Commencer avec MediaTailor](#).

Étape 1 : Création d'une fonction

Au cours de cette étape, vous créez une fonction qui classe le type d'appareil du spectateur en fonction de la chaîne de l'agent utilisateur et enregistre le résultat dans les paramètres du lecteur. La fonction utilise un type de sortie personnalisé (aucun appel d'API externe) avec une [Référence d'expression JSonata](#) expression pour évaluer l'agent utilisateur.

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Fonctions.
3. Choisissez Créer une fonction.
4. Dans le mode de l'assistant de création, sélectionnez Créer à partir de zéro, puis choisissez Continuer.
5. Sous Type de fonction, sélectionnez la vignette de sortie personnalisée.
6. Sous Détails de la fonction, entrez les informations suivantes :
 - Identifiant de la fonction : `myFirstFunction`
 - Description : `Classify device type from user agent`
7. Sous Configuration de sortie personnalisée, dans la section Sortie, ajoutez une ligne :
 - Clé : `player_params.deviceType`
 - Value (Valeur) : `{% $contains(session.user_agent, 'CTV') ? 'ctv' : $contains(session.user_agent, 'Mobile') ? 'mobile' : 'desktop' %}`
8. Choisissez Créer une fonction.

Une notification de réussite confirme que la fonction a été créée et vous êtes redirigé vers la page de détails de la fonction.

La configuration de fonction qui en résulte est la suivante :

```
{
  "FunctionId": "myFirstFunction",
  "FunctionType": "CUSTOM_OUTPUT",
  "Description": "Classify device type from user agent",
```

```
"CustomOutputConfiguration": {
  "Runtime": "JSONATA",
  "Output": {
    "player_params.deviceType": "{% $contains(session.user_agent, 'CTV') ?
'ctv' : $contains(session.user_agent, 'Mobile') ? 'mobile' : 'desktop' %}"
  }
}
```

Étape 2 : associer la fonction à une configuration de lecture

Associez la fonction à un crochet du cycle de vie de votre configuration de lecture. Le mappage indique MediaTailor quand exécuter la fonction.

1. Dans le volet de navigation, sélectionnez Configurations.
2. Choisissez la configuration de lecture que vous souhaitez mettre à jour.
3. Choisissez Modifier.
4. Développez la section Configuration des fonctions.
5. Pour le hook d'initialisation de session, sélectionnez-le dans le myFirstFunction menu déroulant.
6. Choisissez Enregistrer.

Cela se rattache myFirstFunction au crochet du [Pre-session initialisation](#) cycle de vie. Le mappage des fonctions qui en résulte est le suivant :

```
{
  "FunctionMapping": {
    "PRE_SESSION_INITIALIZATION": "myFirstFunction"
  }
}
```

MediaTailor exécute la fonction une fois au début de chaque nouvelle session sur cette configuration de lecture.

Étape 3 : démarrer une session et vérifier que la fonction a été exécutée

Démarrez une nouvelle session de lecture pour déclencher la fonction. Envoyez une demande d'initialisation de session au point de terminaison d'initialisation de session de votre configuration de lecture.

MediaTailor publie automatiquement CloudWatch des métriques pour chaque exécution de fonction, sans inscription. Après avoir démarré une session, vérifiez les métriques suivantes dans l'espace de AWS/MediaTailor noms pour confirmer que votre fonction a été exécutée :

- `PreSessionInitHook.Invocations`— Confirme que l'hameçon a été activé.
- `PreSessionInitHook.Errors`— Doit être égal à 0 en cas de réussite de la fonction.
- `Function.Invocations`— Confirme la fonction individuelle exécutée. Cette métrique inclut `FunctionId`, `FunctionType`, et les `HookType` dimensions afin que vous puissiez filtrer `myFirstFunction` spécifiquement.

Si la fonction échoue, MediaTailor envoie les événements du journal d'erreurs dans Manifest Logs par défaut (aucune configuration n'est requise) :

- `PRE_SESSION_INIT_HOOK_ERROR`— Hook-level échec avec `errorType` et `cause`.
- `PRE_SESSION_INIT_FUNCTION_ERROR`— Function-level échec avec les détails spécifiques `functionId` et les détails de l'erreur.

L'exemple suivant montre un `PRE_SESSION_INIT_FUNCTION_ERROR` événement lié à une erreur de syntaxe dans l'expression de la fonction :

```
{
  "eventTimestamp": "2024-01-01T12:00:00.076000000Z",
  "eventType": "PRE_SESSION_INIT_FUNCTION_ERROR",
  "eventDescription": "Function execution failed",
  "awsAccountId": "123456789012",
  "originId": "my-config",
  "sessionId": "session-123",
  "requestId": "req-abc",
  "eventId": "5dc6f040-0f72-4e8c-a64e-25eeef62708c",
  "functionId": "myFirstFunction",
  "functionType": "CUSTOM_OUTPUT",
  "executionTimeMs": 2,
  "errorType": "SYNTAX_ERROR",
  "cause": "Expected \"\\\" before end of expression",
  "input": {}
}
```

Utilisez le `eventId` champ pour corréliser les événements d'erreur de type hook et fonction pour une même exécution. Le `errorType` champ indique la classe d'échec. Consultez [Résolution des problèmes et surveillance](#) la liste complète des types d'erreurs et des correctifs.

Note

Pour une journalisation détaillée des réussites, inscrivez-vous à votre configuration Manifest Log `PRE_SESSION_INIT_HOOK_SUMMARY` et enregistrez les `PRE_SESSION_INIT_FUNCTION_COMPLETED` événements dans cette configuration. Les événements récapitulatifs indiquent le résultat du hook pour chaque exécution. Les événements terminés affichent les request/response informations relatives à l'entrée, à la sortie et au protocole HTTP de chaque fonction. Ils sont désactivés par défaut afin de minimiser les coûts de journalisation. Pour de plus amples informations, veuillez consulter [Résolution des problèmes et surveillance](#).

Que se passe-t-il dans les coulisses

Voici le flux de requêtes complet pour la fonction que vous venez de créer :

1. Le joueur lance une session avec MediaTailor.
2. MediaTailor déclenche le `PRE_SESSION_INITIALIZATION` cycle de vie et s'exécute `myFirstFunction`.
3. La fonction évalue le `session.user_agent` champ et écrit `ctvmobile`, ou `desktop` dans `player_params.deviceType`
4. MediaTailor crée la session et renvoie le manifeste au joueur.
5. Le joueur rencontre une interruption publicitaire pendant la lecture.
6. MediaTailor déclenche le hook `PRE_ADS_REQUEST` du cycle de vie, puis construit la demande ADS. Comme il `deviceType` est stocké dans les paramètres du joueur, il peut être inclus dans l'URL de demande ADS par le biais d'une substitution dynamique de variables.
7. L'ADS utilise le type d'appareil pour renvoyer des créations publicitaires ciblées.
8. MediaTailor insère les publicités dans le manifeste et le renvoie au joueur.

Si la fonction échoue pour une raison quelconque, MediaTailor la sortie est supprimée et procède comme si aucune fonction n'était attachée. Le spectateur voit toujours des publicités, mais sans le ciblage par type d'appareil.

Sujets suggérés

Une fonction fonctionnelle est désormais associée à une configuration de lecture. À partir d'ici :

- Pour savoir quels champs d'entrée et quels espaces de noms de sortie sont disponibles à chaque hook du cycle de vie, consultez [Hooks de cycle de vie](#).
- Pour en savoir plus sur les différents types de fonctions et sur la façon de les enchaîner, voir [Types de fonctions et composition](#).
- Pour consulter des exemples pratiques complets, voir [Exemples de fonctions](#).

Fonctions, cycle de vie, crochets

Un hook de cycle de vie définit à MediaTailor quel moment votre fonction est exécutée pendant la lecture. Cette page est une référence complète pour les champs de saisie, les espaces de noms de sortie et les règles qui régissent le flux de données à chaque hook.

Présentation de

MediaTailor prend en charge deux hooks de cycle de vie :

- **PRE_SESSION_INITIALIZATION** se déclenche une fois lorsqu'un spectateur démarre une nouvelle session. Utilisez-le pour des tâches de configuration ponctuelles, telles que la récupération de segments d'audience. À ce stade, aucune interruption publicitaire ne s'est produite, le contexte de la pause publicitaire n'est donc pas disponible.
- **PRE_ADS_REQUEST** se déclenche avant chaque requête du serveur de décision publicitaire (ADS), soit une fois par pause publicitaire dans le flux. Utilisez-le pour personnaliser la demande ADS avec des données de ciblage, modifier l'URL ADS ou ajouter des en-têtes.

La principale différence réside dans le timing : PRE_SESSION_INITIALIZATION s'exécute une seule fois et configure des données qui persistent pendant toute la durée de la session, tandis qu'elles PRE_ADS_REQUEST s'exécutent de manière répétée et permet d'adapter chaque demande ADS à une pause publicitaire spécifique.

Pre-session initialisation

Quand ça se déclenche

MediaTailor exécute la fonction mappée `PRE_SESSION_INITIALIZATION` une fois, au début d'une nouvelle session de lecture. La fonction s'exécute avant de MediaTailor construire la réponse manifeste initiale.

Input

`session.*`, `player_params.*` et `event.*`. Pour tous les champs disponibles, consultez [Référence du champ de saisie](#).

Espace de noms de sortie autorisé

Namespace	Types acceptés
<code>player_params.*</code>	Chaînes, nombres, booléens

Les valeurs écrites `player_params.*` sont conservées dans la session. Ils sont disponibles :

- En tant que saisie dans le hook `PRE_ADS_REQUEST` du cycle de vie via `player_params.*`
- Dans ADS, demandez des URL via [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#) (par exemple, `[player_params.deviceType]`)
- Pendant toute la durée de la session pendant toutes les pauses publicitaires

Note

La taille sérialisée totale de toutes les clés et valeurs de `player_params` sortie ne doit pas dépasser 1 000 caractères. Si le total dépasse cette limite, la sortie de la fonction est ignorée. Pour de plus amples informations, veuillez consulter [Limites des fonctions](#).

Cas d'utilisation typiques

- Récupérez les données d'identité ou d'audience auprès d'un service externe et stockez-les dans les paramètres du lecteur pour les utiliser lors de requêtes ADS ultérieures.

- Classez le type d'appareil en fonction de l'agent utilisateur et écrivez la classification dans un paramètre du joueur.
- Définissez les valeurs de paramètres de lecteur par défaut sur lesquelles repose le traitement des pauses publicitaires en aval.
- Stockez les valeurs dans les paramètres du lecteur qui sont inclus dans l'URL ADS via [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

Comportement de défaillance

Si une fonction associée `PRE_SESSION_INITIALIZATION` échoue pour une raison quelconque, MediaTailor supprime la sortie de la fonction et procède comme si aucune fonction n'était attachée. La session démarre normalement sans les valeurs des paramètres du lecteur de la fonction.

Pre-ads demande

Quand ça se déclenche

MediaTailor exécute la fonction mappée à `PRE_ADS_REQUEST` une fois par pause publicitaire, juste avant d'envoyer la demande à l'ADS. La fonction s'exécute chaque fois qu'une opportunité publicitaire est rencontrée pendant le traitement du manifeste.

Input

Tous les champs provenant de `PRE_SESSION_INITIALIZATION` `avail.*`, `plusscte.*`, `asset.*`, et `adsRequest.*` (URL, méthode, en-têtes, corps). Pour tous les champs disponibles, consultez [Référence du champ de saisie](#).

Espace de noms de sortie autorisé

Namespace	Types acceptés	Comment la sortie est utilisée
<code>player_params.*</code>	Chaînes, nombres, booléens	Remplace les paramètres du lecteur de session pour cette pause publicitaire. Accessible à l'URL de la demande ADS via MediaTailor variables publicitaires dynamiques pour les demandes ADS .

Namespace	Types acceptés	Comment la sortie est utilisée
<code>session.*</code>	Chaînes, nombres, booléens	Remplace les variables de session pour cette pause publicitaire. Accessible à l'URL de la demande ADS par substitution dynamique de variables.
<code>avail.*</code>	Chaînes, nombres, booléens	Remplace les variables disponibles pour cette pause publicitaire. Accessible à l'URL de la demande ADS par substitution dynamique de variables.
<code>scte.*</code>	Chaînes, nombres, booléens	Remplace les variables SCTE pour cette pause publicitaire. Accessible à l'URL de la demande ADS par substitution dynamique de variables.
<code>asset.*</code>	Chaînes, nombres, booléens	Remplace les variables de métadonnées des actifs (issues des EXT-X-ASSET balises) pour cette pause publicitaire. Accessible à l'URL de la demande ADS par substitution dynamique de variables. Pour plus d'informations sur les marqueurs publicitaires HLS, consultez la section Marqueurs publicitaires HLS .

Namespace	Types acceptés	Comment la sortie est utilisée
<code>adsRequest.*</code>	Chaîne	Remplace la demande ADS pour cette pause publicitaire uniquement. Champs pris en charge : <code>urlmethod,headers.<name>,,body</code> . La <code>url</code> valeur est traitée comme un modèle et est prise en charge MediaTailor variables publicitaires dynamiques pour les demandes ADS après l'exécution de la fonction. N'a pas persisté.

Note

Toutes les sorties du PRE_ADS_REQUEST hook sont des remplacements transitoires : elles s'appliquent uniquement à la demande ADS de la pause publicitaire en cours et ne sont pas conservées pendant la session.

Exemple : réécriture de la requête ADS :

```
{
  "Output": {
    "adsRequest.url": "${'https://ads.example.com/v1/vast?sid=' & session.id &
'&genre=' & player_params.genre%}",
    "adsRequest.headers.X-Custom-Token": "${player_params.auth_token%}"
  }
}
```

Cela remplace l'URL de l'ADS et ajoute un en-tête personnalisé pour la pause publicitaire en cours.

Cas d'utilisation typiques

- Réécrivez l'URL de la demande ADS pour acheminer le trafic entre les différents serveurs publicitaires à des fins de A/B test.
- Ajoutez des données d'enrichissement (segments d'audience, jetons d'identité) à l'URL ou aux en-têtes de la demande ADS.
- Modifiez de manière conditionnelle les paramètres de demande ADS en fonction des données de SCTE-35 signal ou de l'indice de disponibilité.

Comportement de défaillance

Si une fonction associée PRE_ADS_REQUEST échoue pour une raison quelconque, MediaTailor supprime la sortie de la fonction et procède comme si aucune fonction n'était attachée. La demande ADS est envoyée en utilisant les paramètres de session et de demande d'origine sans modification.

Référence du champ de saisie

Champ	Type	INITIALISATION_PRÉ SESSION	DEMANDE_P RE_ANNONCE
session.id	Long	✓	✓
session.uuid	Chaîne	✓	✓
session.c lient_ip	Chaîne	✓	✓
session.u ser_agent	Chaîne	✓	✓
session.referer *	Chaîne	✓	✓
session.a vail_dura tion_secs	Long	✗	✓

Champ	Type	INITIALISATION_PRÉ SESSION	DEMANDE_P RE_ANNONCE
session.avail_duration_ms	Long	✗	✓
player_params.*	Chaîne	✓	✓
event.id	Chaîne	✓	✓
event.hook	Chaîne	✓	✓
event.timestamp	Chaîne	✓	✓
avail.index	Int	✗	✓
avail.random	Long	✗	✓
avail.source_content_time_epoch_ms	Long	✗	✓
scte.event_id	Int	✗	✓
scte.avail_num	Int	✗	✓
scte.segmentation_event_id	Int	✗	✓
scte.segmentation_type_id	Int	✗	✓
scte.segmentation_upid	Chaîne	✗	✓

Champ	Type	INITIALISATION_PRÉ SESSION	DEMANDE_P RE_ANNONCE
scte.segm entation_ upid.assetId	Chaîne	x	✓
scte.segm entation_ upid.cueD ata.key	Chaîne	x	✓
scte.segm entation_ upid.cueD ata.value	Chaîne	x	✓
scte.uniq ue_program_id	Int	x	✓
scte.arch ive_allow ed_flag	Booléen	x	✓
scte.deliv ery_not_ restricte d_flag	Booléen	x	✓
scte.devi ce_restri ctions	Int	x	✓
scte.no_r egional_b lackout_flag	Booléen	x	✓

Champ	Type	INITIALISATION_PRÉ SESSION	DEMANDE_P RE_ANNONCE
<code>scte.segmen t_num</code>	Int	X	✓
<code>scte.segm ents_expected</code>	Int	X	✓
<code>scte.sub_ segment_num</code>	Int	X	✓
<code>scte.sub_ segments_ expected</code>	Int	X	✓
<code>scte.avai ls_expected</code>	Long	X	✓
<code>asset.*</code>	Chaîne	X	✓
<code>adsRequest.url</code>	Chaîne	X	✓
<code>adsReques t.method</code>	Chaîne	X	✓
<code>adsReques t.headers .<key></code>	Chaîne	X	✓
<code>adsRequest.body</code>	Chaîne	X	✓

* n' `session.referer` est présent que lorsqu'un en-tête Referer est inclus dans la demande d'initialisation de session. `$exists(session.referer)` À utiliser pour vérifier avant d'accéder.

Types de fonctions et composition

AWS Elemental MediaTailor propose quatre types de fonctions, chacune étant conçue pour une catégorie de travail différente. Vous choisissez un type lorsque vous créez une fonction, et ce type

détermine ce que la fonction peut faire lors de l'exécution. Cette page explique comment chaque type s'exécute, quand l'utiliser et comment composer des fonctions dans des pipelines en plusieurs étapes.

Vue d'ensemble des types de fonctions

Type	Catégorie	Objectif
CUSTOM_OUTPUT	Transformation de données	Évaluez les expressions par rapport à l'état actuel de la session et produisez des résultats. Aucun appel externe.
HTTP_REQUEST	Intégration externe	Passez un appel HTTP à un service externe, puis évaluez les expressions de sortie qui peuvent faire référence à la réponse.
SEQUENTIAL_EXECUTOR	Orchestration	Exécutez une séquence de fonctions dans l'ordre, en transmettant les données entre les étapes via des données temporaires.
CONCURRENT_EXECUTOR	Orchestration	Exécutez un ensemble de fonctions en parallèle jusqu'à une simultanéité maximale, puis combinez leurs sorties.

Chaque type joue un rôle distinct. Choisissez un type lorsque vous créez une fonction. Ce type détermine ce que la fonction peut faire au moment de l'exécution.

Sortie personnalisée

Quand l'utiliser

À utiliser CUSTOM_OUTPUT lorsque la logique de votre fonction est purement informatique. Les cas d'utilisation courants incluent la classification des appareils en fonction des chaînes d'agent utilisateur, la sélection des URL du serveur de décision publicitaire avec une logique de A/B test et le reformatage des paramètres du lecteur avant qu'ils n'atteignent le serveur de décision publicitaire.

CUSTOM_OUTPUT Les fonctions sont le type le plus rapide car elles évaluent les expressions en mémoire sans appel réseau. Choisissez ce type chaque fois que vous pouvez atteindre votre objectif sans contacter un service externe.

Champs de configuration

Une CUSTOM_OUTPUT fonction comporte deux champs :

- Runtime : langage d'expression. Réglez ce paramètre sur JSONATA.
- Sortie : carte des paires clé-valeur. Chaque clé est un nom de liaison de sortie (tel que `player_params.device_type`), et chaque valeur est une expression qui est MediaTailor évaluée lors de l'exécution.

Note

Permet d'`{%...%}` encapsuler des expressions JSonata dans des champs de chaînes. Les valeurs sans ces délimiteurs sont traitées comme des chaînes statiques. Par exemple, `"{%session.id%}"` évalue l'expression alors que `"session.id"` c'est le texte littéral.

Comportement d'exécution

Lors de l' MediaTailor exécution d'une CUSTOM_OUTPUT fonction, elle évalue chaque expression du bloc de sortie par rapport à l'état actuel de la session. Toutes les expressions voient le même instantané d'entrée. Les résultats évalués deviennent les sorties de la fonction.

Si une expression ne parvient pas à être évaluée, MediaTailor supprime l'intégralité de la sortie de la fonction et procède comme si aucune fonction n'était attachée. La fonction ne valide pas partiellement les résultats.

Exemple : répartition A/B du trafic

La fonction suivante attribue de manière aléatoire chaque pause publicitaire à l'une des deux URL du serveur de décision publicitaire. Il est conçu pour répondre aux exigences du PRE_ADS_REQUEST cycle de vie.

```
{
  "FunctionId": "abTestAdsUrl",
  "FunctionType": "CUSTOM_OUTPUT",
  "CustomOutputConfiguration": {
    "Runtime": "JSONATA",
    "Output": {
      "adsRequest.url": "{%$random() < 0.5 ? 'https://ads.example.com/v1' : 'https://ads.example.com/v2'%}"
    }
  }
}
```

Pour une présentation complète de cet exemple, consultez [Exemples de fonctions](#).

Requête HTTP

Quand l'utiliser

À utiliser HTTP_REQUEST lorsque votre fonction doit appeler un service externe. Les cas d'utilisation courants incluent la récupération de données d'identité auprès d'un fournisseur de résolution, la récupération de segments d'audience à partir d'une plateforme de gestion de données et l'envoi d'informations de session à un terminal de journalisation.

Champs de configuration

Une HTTP_REQUEST fonction comporte les champs suivants :

- **Runtime** : langage d'expression. Réglez ce paramètre sur JSONATA.
- **MethodType**— La méthode HTTP. Les valeurs prises en charge sont GET et POST.
- **URL** : URL à laquelle envoyer la demande. Vous pouvez utiliser une URL statique ou une expression JSonata qui génère l'URL de manière dynamique.
- **En-têtes** : en-têtes HTTP à inclure dans la demande, spécifiés sous forme de paires nom d'en-tête et valeur. Utilisez la syntaxe des {%...}% expressions pour les valeurs d'en-tête dynamiques. Les valeurs statiques peuvent être spécifiées directement sous forme de chaînes.

- **Corps** : corps de la demande à envoyer. Utilisé pour les POST demandes. Vous pouvez utiliser une expression JSonata pour construire le corps de manière dynamique.
- **RequestTimeoutMilliseconds(obligatoire)** — Combien de temps faut-il attendre pour obtenir une réponse ?
- **Sortie** : définit les valeurs à produire une fois l'appel HTTP terminé. Chaque entrée associe une clé de sortie (telle que `player_params.envelope_id`) à une expression qui peut faire référence à l'`response` objet.

Pour connaître les limites de taille et les restrictions qui s'appliquent à ces champs, consultez [Restrictions](#).

Comment la demande est traitée

MediaTailor traite une `HTTP_REQUEST` fonction en deux étapes :

1. **Génère la demande** : MediaTailor évalue les `Body` expressions `UrlHeaders`, et par rapport à l'état actuel de la session. Ces valeurs évaluées constituent la requête HTTP sortante.
2. **Traiter la réponse** — Une fois l'appel HTTP terminé, MediaTailor évalue les expressions du bloc de sortie. Ces expressions peuvent faire référence à la fois à l'état de session d'origine et à l'`response` objet renvoyé par l'appel.

Champs de réponse

Une fois l'appel HTTP terminé, vous pouvez référencer les champs suivants dans vos expressions de sortie :

Champ	Type	Description
<code>response.body</code>	Objet ou tableau	Le corps de la réponse est analysé au format JSON. Défini sur <code>null</code> si le corps dépasse 20 000 caractères ou s'il ne s'agit pas d'un JSON valide.
<code>response.statusCode</code>	Entier	Code d'état HTTP renvoyé par le service externe. Réglez

Champ	Type	Description
		<code>null</code> sur en cas de défaillance du réseau.
<code>response.text</code>	Chaîne	Le corps brut de la réponse sous forme de chaîne, tronqué à 20 000 caractères. Réglez "Internal Error" sur en cas de défaillance du réseau.

 Important

Ce `response.body` champ est utilisé `null` lorsque la réponse dépasse 20 000 caractères, même si la réponse est un JSON valide.

 Note

L'objet de réponse est disponible uniquement dans le bloc de sortie d'une `HTTP_REQUEST` fonction. Vous ne pouvez pas référencer les champs de réponse dans les champs URL, en-têtes ou corps. Dans `aSEQUENTIAL_EXECUTOR`, chaque `HTTP_REQUEST` fonction ne peut accéder qu'à sa propre réponse.

La valeur de `null` signifie que les données ne sont pas disponibles. Cela se produit lorsque l'appel HTTP échoue (erreur réseau ou délai d'attente) ou lorsque le corps de la réponse dépasse 20 000 caractères ou n'est pas un JSON valide.

Comportement des défaillances du réseau

Si l'appel HTTP échoue en raison d'une erreur réseau ou d'un délai d'attente, `response.statusCode` et si vous `response.body` êtes défini sur `null`, et `response.text` est défini sur "Internal Error" Vos expressions de sortie s'exécutent toujours. Vérifiez donc toujours `response.statusCode` avant d'utiliser les données de réponse.

i Tip

Utilisez une expression conditionnelle pour gérer les échecs avec élégance :

```
{%response.statusCode = 200 ? response.body.value : 'default'%}
```

Exemple : récupérer les données d'identité

La fonction suivante appelle une API de résolution d'identité au début de la session et enregistre le résultat dans les paramètres du joueur. Il est conçu pour répondre aux exigences du PRE_SESSION_INITIALIZATION cycle de vie.

```
{
  "FunctionId": "fetchIdentityEnvelope",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://identity.example.com/v1/resolve?ip=' &
$encodeUrlComponent(session.client_ip)%}",
    "Headers": {
      "Authorization": "{%'Bearer my_api_token'%}",
      "Accept": "application/json"
    },
    "RequestTimeoutMilliseconds": 2000,
    "Output": {
      "player_params.identity_envelope": "{%response.statusCode = 200 ?
response.body.envelope : '%%}"
    }
  }
}
```

Pour une présentation complète d'un exemple similaire, consultez [Exemples de fonctions](#).

Exécuteur séquentiel

Quand l'utiliser

A SEQUENTIAL_EXECUTOR exécute une série de fonctions une par une, dans l'ordre. Chaque étape peut utiliser les résultats des étapes précédentes.

SEQUENTIAL_EXECUTOR À utiliser lorsque votre logique nécessite plusieurs étapes qui dépendent des résultats des uns et des autres. Les cas d'utilisation courants incluent la récupération de données d'identité, puis leur utilisation pour récupérer des segments d'audience, l'exécution d'une étape de classification, puis l'appel conditionnel de différents services externes en fonction du résultat, et la création d'URL de demandes d'annonces complexes à partir de plusieurs sources de données.

Champs de configuration

Une **SEQUENTIAL_EXECUTOR** fonction comporte les champs suivants :

- **Runtime** : langage d'expression. Réglez ce paramètre sur **JSONATA**.
- **FunctionList**— Une liste ordonnée de 1 à 10 étapes. Chaque étape indique **FunctionId** la fonction à exécuter. Vous pouvez éventuellement ajouter une **RunCondition** expression pour contrôler si l'étape est exécutée ou ignorée.
- **Sortie** : définit les valeurs à produire une fois toutes les étapes terminées. Chaque entrée associe une clé de sortie (telle que `player_params.envelope`) à une expression qui peut faire référence à des données produites par n'importe quelle étape de la séquence. En cas d'omission, toutes les sorties des différentes fonctions de la séquence sont utilisées.
- **TimeoutMilliseconds(obligatoire)** — Durée maximale nécessaire à la fin de l'ensemble de la séquence. Si la séquence dépasse ce délai, MediaTailor supprime toutes les sorties de la séquence.

Exécution ordonnée et flux de données

MediaTailor exécute chaque étape de la séquence de la première à la dernière. Une fois chaque étape terminée, les valeurs produites sont fusionnées dans un ensemble de résultats. Les étapes suivantes peuvent accéder aux données de session d'origine ainsi qu'à toutes les valeurs produites par les étapes précédentes.

Les données temporaires constituent le principal mécanisme de transmission des données entre les étapes. Lorsqu'une fonction écrit sur une `temp.*` touche, l'étape suivante permet de lire cette valeur. Les paramètres du joueur et les champs de demande d'annonces écrits lors des étapes précédentes sont également visibles lors des étapes suivantes.

 Note

Les données temporaires acceptent tous les types de données, y compris les objets et les tableaux. Les paramètres du lecteur et les champs de demande d'annonces n'acceptent que les chaînes, les nombres, les booléens et les valeurs nulles).

Per-step conditions de course

Chaque étape de la séquence comporte un `RunCondition` champ facultatif. Ce champ contient une expression qui renvoie `true` ou `false`. MediaTailor évalue l'`RunCondition` expression immédiatement avant d'exécuter cette étape.

Si l'`RunCondition` expression est évaluée à `false`, MediaTailor saute complètement l'étape et passe à la suivante. Si le `RunCondition` champ est omis, l'étape est toujours exécutée.

```
{ "FunctionId": "retryFetch", "RunCondition": "{%temp.statusCode = 500%}" }
```

Ce mécanisme vous permet de créer des pipelines conditionnels. Par exemple, vous pouvez exécuter une recherche d'identité à l'étape 1, puis exécuter une recherche de segment de manière conditionnelle à l'étape 2 uniquement si l'étape 1 a renvoyé une identité valide.

Comment fonctionne le bloc de sortie

Le bloc de sortie sur `SEQUENTIAL_EXECUTOR` contrôle ce que produit la séquence une fois toutes les étapes terminées :

- **Bloc de sortie présent** : MediaTailor évalue les expressions du bloc de sortie par rapport à l'état cumulé final et enregistre uniquement ces sorties. Toutes les sorties produites par les étapes précédentes qui ne sont pas référencées dans le bloc de sortie séquentiel sont supprimées.
- **Bloc de sortie absent** : MediaTailor enregistre directement toutes les sorties accumulées de toutes les étapes.

 Tip

Omettez le bloc de sortie lorsque vous souhaitez que la sortie de chaque fonction passe. Ajoutez un bloc de sortie lorsque vous devez filtrer, renommer ou transformer les résultats cumulés avant de les enregistrer.

Configuration du délai d'attente

Le `TimeoutMilliseconds` champ définit une date limite pour l'ensemble de la séquence. Ce délai couvre toutes les étapes, y compris les appels HTTP effectués par les fonctions. Si la séquence dépasse le délai d'attente, MediaTailor supprime toutes les sorties de la séquence et procède comme si aucune fonction n'était attachée.

Les `HTTP_REQUEST` fonctions individuelles respectent toujours leur propre `RequestTimeoutMilliseconds` réglage. Le délai d'attente de la séquence agit comme une limite extérieure qui limite le temps d'exécution total.

Exemple : Réessayez en cas d'échec HTTP

Cet exemple appelle une API d'identité et réessaie automatiquement si le premier appel renvoie une erreur de serveur. Il utilise deux fonctions `HTTP_REQUEST` orchestrées par un `SEQUENTIAL_EXECUTOR`.

Étape 1 — Récupération principale (**fetchIdentity**) :

```
{
  "FunctionId": "fetchIdentity",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://identity.example.com/v1/resolve?ip=' & session.client_ip%}'",
    "Headers": {
      "Accept": "application/json"
    },
  },
  "RequestTimeoutMilliseconds": 1000,
  "Output": {
    "temp.statusCode": "{%response.statusCode%}",
    "temp.envelope": "{%response.statusCode = 200 ? response.body.envelope :
null%}"
  }
}
```

Étape 2 — Réessayez en cas d'échec (**retryIdentity**) :

```
{
```



```

    "FunctionId": "retryIdentity",
    "FunctionType": "HTTP_REQUEST",
    "HttpRequestConfiguration": {
        "Runtime": "JSONATA",
        "MethodType": "GET",
        "Url": "{%'https://identity-fallback.example.com/v1/resolve?ip=' &
session.client_ip%}",
        "Headers": {
            "Accept": "application/json"
        },
        "RequestTimeoutMilliseconds": 1000,
        "Output": {
            "temp.statusCode": "{%response.statusCode%}",
            "temp.envelope": "{%response.statusCode = 200 ? response.body.envelope :
null%}"
        }
    }
}

```

Séquence (**identityWithRetry**) :

```

{
    "FunctionId": "identityWithRetry",
    "FunctionType": "SEQUENTIAL_EXECUTOR",
    "SequentialExecutorConfiguration": {
        "Runtime": "JSONATA",
        "TimeoutMilliseconds": 2000,
        "FunctionList": [
            { "FunctionId": "fetchIdentity" },
            { "FunctionId": "retryIdentity", "RunCondition": "{%temp.statusCode >=
500%}" }
        ],
        "Output": {
            "player_params.envelope": "{%temp.envelope%}"
        }
    }
}

```

Comment ça fonctionne :

1. `fetchIdentity` appelle l'API d'identité et écrit le code d'état et l'enveloppe dans `temp.*`.

2. Si le code d'état est supérieur ou égal à 500, `RunCondition` l'étape 2 est évaluée `true` et `retryIdentity` exécutée. Il écrase `temp.statusCode` et `temp.envelope` avec la réponse de nouvelle tentative.
3. Si le premier appel a réussi, l'étape 2 est ignorée.
4. La séquence dans laquelle le bloc `temp.envelope` de sortie écrit `player_params.envelope`.

Exécuteur testamentaire simultané

Quand l'utiliser

A `CONCURRENT_EXECUTOR` exécute un ensemble de fonctions enfants en parallèle jusqu'à une simultanéité maximale, puis combine leurs sorties une fois que tous les enfants ont terminé.

`CONCURRENT_EXECUTOR` à utiliser lorsque votre logique nécessite plusieurs opérations indépendantes qui ne dépendent pas des résultats des unes des autres. Les cas d'utilisation courants incluent la réalisation d'appels HTTP parallèles vers différents services d'enrichissement, l'exécution simultanée de recherches de données indépendantes et la réduction de la latence totale en superposant les requêtes réseau.

Champs de configuration

Une `CONCURRENT_EXECUTOR` fonction comporte les champs suivants :

- **Runtime** : langage d'expression. Réglez ce paramètre sur `JSONATA`.
- **FunctionList**— Une liste de 1 à 10 enfants à courir en parallèle. Chaque entrée indique `FunctionId` la fonction à exécuter. Vous pouvez éventuellement ajouter une `RunCondition` expression pour contrôler si l'enfant court ou s'il est ignoré. Vous pouvez également ajouter un `Alias` pour donner à l'enfant un nom unique au sein de l'exécuteur testamentaire.
- **Sortie** : définit les valeurs à produire une fois que tous les enfants ont terminé. Chaque entrée associe une clé de sortie (telle que `player_params.device_type`) à une expression qui peut faire référence à des données produites par n'importe quel enfant de l'exécuteur.
- **MaxConcurrency**— Le nombre maximum d'enfants pouvant courir simultanément. Plage valide : 1 —2. Valeur par défaut : 2.
- **TimeoutMilliseconds(obligatoire)** — Durée maximale pendant laquelle tout l'exécuteur testamentaire doit terminer. Si l'exécuteur dépasse ce délai, MediaTailor supprime toutes les sorties de l'exécuteur. Valeur par défaut : 2000.

Exécution parallèle et flux de données

MediaTailor fait démarrer les enfants en même temps. Lorsque la liste de fonctions contient plus d'entrées que `MaxConcurrency`, MediaTailor démarre les enfants supplémentaires au fur et à mesure que ceux en cours d'exécution sont terminés. Le nombre d'enfants qui courent simultanément reste égal ou inférieur à la limite de simultanéité.

Le résultat de chaque enfant est stocké dans son espace de noms : `explicitAlias`, ou celui de l'enfant `FunctionId` si aucun alias n'est défini. Une fois que tous les enfants ont terminé, les `Output` expressions de l'exécuteur testamentaire peuvent faire référence aux résultats de n'importe quel enfant à l'aide du chemin `function.<namespace>.<path>`.

Note

Les espaces de noms doivent être uniques pour tous les enfants de l'exécuteur testamentaire. MediaTailor valide cette exigence au moment de la création.

Per-child conditions de course

Chaque enfant de l'exécuteur testamentaire possède un `RunCondition` champ facultatif. Ce champ contient une expression qui renvoie `true` ou `false`. MediaTailor évalue l'`RunCondition` expression juste avant de démarrer cet enfant.

Si la valeur de l'`RunCondition` expression est égale à `false`, le nom de l'enfant MediaTailor est complètement ignoré. Si le `RunCondition` champ est omis, l'enfant court toujours.

```
{ "FunctionId": "fetchDevice", "RunCondition": "{%session.player_params.device_id != null%}" }
```

Avec ce mécanisme, vous pouvez inclure des enfants de manière conditionnelle. Par exemple, vous pouvez ignorer une recherche d'appareil lorsque l'identifiant de l'appareil n'est pas présent dans la session.

Comment fonctionne le bloc de sortie

Le bloc de sortie sur `CONCURRENT_EXECUTOR` contrôle ce que l'exécuteur produit une fois que tous les enfants ont terminé. Chaque entrée associe une clé de sortie à une expression qui peut faire référence aux résultats de n'importe quel enfant à l'aide de `{%function.<namespace>.<path>%}`.

Par exemple, si un enfant a un alias de `fetchUser` et écrit `danstemp.user`, l'expression de sortie de l'exécuteur peut le référencer comme `{%function.fetchUser.temp.user%}`.

Configuration du délai d'attente

Le `TimeoutMilliseconds` champ fixe une date limite pour l'ensemble de l'exécuteur testamentaire. Ce délai d'attente couvre tous les enfants, y compris tous les appels HTTP effectués par les fonctions. Si l'exécuteur dépasse le délai d'attente, MediaTailor supprime toutes les sorties de l'exécuteur et procède comme si aucune fonction n'était attachée.

Les `HTTP_REQUEST` fonctions individuelles respectent toujours leur propre `RequestTimeoutMilliseconds` réglage. Le délai d'expiration de l'exécuteur agit comme une limite extérieure qui limite le temps d'exécution total.

Exemple : appels d'enrichissement parallèles

Cet exemple appelle deux API d'enrichissement en parallèle et combine leurs résultats dans les paramètres du lecteur. Il utilise deux fonctions `HTTP_REQUEST` orchestrées par un `CONCURRENT_EXECUTOR`.

Enfant 1 — Profil utilisateur (**fetchUserProfile**)

```
{
  "FunctionId": "fetchUserProfile",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "Url": "{%'https://enrichment.example.com/user/' &
session.player_params.user_id%}",
    "Headers": {},
    "Output": {
      "temp.user_segment": "{%response.body.segment%}"
    }
  }
}
```

Enfant 2 — Informations sur l'appareil (**fetchDeviceInfo**)

```
{
  "FunctionId": "fetchDeviceInfo",
  "FunctionType": "HTTP_REQUEST",
```

```

    "HttpRequestConfiguration": {
      "Runtime": "JSONATA",
      "Url": "{% 'https://enrichment.example.com/device/' &
session.player_params.device_id%}",
      "Headers": {},
      "Output": {
        "temp.device_category": "{%response.body.category%}"
      }
    }
  }
}

```

Exécuteur testamentaire () **parallelEnrichment**

```

{
  "FunctionId": "parallelEnrichment",
  "FunctionType": "CONCURRENT_EXECUTOR",
  "ConcurrentExecutorConfiguration": {
    "Runtime": "JSONATA",
    "MaxConcurrency": 2,
    "TimeoutMilliseconds": 1500,
    "FunctionList": [
      {
        "FunctionId": "fetchUserProfile",
        "Alias": "userCall"
      },
      {
        "FunctionId": "fetchDeviceInfo",
        "Alias": "deviceCall",
        "RunCondition": "{%session.player_params.device_id != null%}"
      }
    ],
    "Output": {
      "player_params.user_segment": "{%function.userCall.temp.user_segment%}",
      "player_params.device_category":
        "{%function.deviceCall.temp.device_category%}"
    }
  }
}

```

Dans cet exemple :

1. MediaTailor démarre les deux enfants simultanément (MaxConcurrency soit 2).
2. L'deviceCall enfant ne court que s'device_id est présent (RunCondition).

3. Une fois les deux appels terminés (dans les 1 500 ms), le bloc Output écrit les résultats combinés dans `player_params`.
4. Si le délai de 1500 ms est dépassé, MediaTailor supprime toutes les sorties et continue sans les données d'enrichissement.

Règles de composition

MediaTailor impose les limites suivantes à la composition des fonctions :

Règle	Limite
Profondeur de nidification maximale	2 (un exécuteur testamentaire peut contenir des fonctions, mais ces enfants ne peuvent pas être eux-mêmes des exécuteurs testamentaires)
Étapes par séquence	1 à 10
Enfants par exécuteur testamentaire simultané	1 à 10
Simultanéité maximale (exécuteur simultané)	1 à 2
Nombre total d'exécutions de fonctions par hook de cycle de vie	20
Références circulaires	Non autorisé. Une fonction ne peut pas se référencer elle-même, directement ou indirectement.
Existence de la fonction	Toutes les fonctions référencées dans a <code>FunctionList</code> doivent exister avant de créer ou de mettre à jour l'exécuteur parent.
Espaces de noms uniques (exécuteur simultané)	Chaque enfant doit avoir un alias ou un identifiant de fonction unique au sein de l'exécuteur.

Lorsque vous créez une fonction, MediaTailor validez la syntaxe des expressions, recherchez les fonctions restreintes, vérifiez que toutes les fonctions référencées existent et détectez les références

circulaires. Lorsque vous associez une fonction à une configuration de lecture, cela confirme MediaTailor également que toutes les touches de sortie de l'ensemble de l'arbre des fonctions sont compatibles avec le hook de cycle de vie attribué.

Création et gestion de fonctions

Cette page explique comment créer, joindre, mettre à jour et supprimer des fonctions à l'aide de la MediaTailor console. Les fonctions vous permettent de personnaliser le comportement des sessions et les demandes publicitaires en exécutant la logique à des moments clés de la lecture.

Configuration d'une fonction

Création d'une fonction à l'aide de la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Fonctions.
3. Choisissez Créer une fonction.
4. Dans le champ Function ID, entrez un nom unique pour la fonction (par exemple, `fetchIdentity`).
5. Pour Type de fonction `CUSTOM_OUTPUT`, sélectionnez `HTTP_REQUEST`, ou `SEQUENTIAL_EXECUTOR`. La console affiche les champs de configuration pour le type sélectionné.
6. Renseignez les champs de configuration spécifiques au type. Pour une description de chaque type de fonction et de ses champs, consultez [Types de fonctions et composition](#).
7. Choisissez Créer une fonction.

Associer une fonction à une configuration de lecture

Une fonction ne s'exécute que lorsque vous l'associez à une configuration de lecture par le biais d'un mappage de fonctions (lien entre un hook du cycle de vie et une fonction). Le mappage des fonctions indique quel hook du cycle de vie déclenche la fonction.

1. Ouvrez la MediaTailor console.
2. Dans le volet de navigation, sélectionnez Configurations.
3. Choisissez la configuration de lecture que vous souhaitez mettre à jour.
4. Dans la section Cartographie des fonctions, choisissez Modifier.

5. Pour chaque hook du cycle de vie, sélectionnez la fonction à associer :

- Hook d'initialisation de session — Choisissez une fonction à exécuter une fois au début de la session.
- Hook de demande publicitaire — Choisissez une fonction à exécuter avant chaque demande ADS.

6. Choisissez Enregistrer.

Gestion des fonctions

Fonctions de visualisation

Accédez à Fonctions pour voir toutes les fonctions de votre compte. Choisissez le nom d'une fonction pour visualiser sa configuration.

Mettre à jour une fonction existante

Pour mettre à jour une fonction, accédez à Fonctions, choisissez la fonction et modifiez la configuration. La mise à jour remplace la définition complète de la fonction.

Tip

Enregistrez une copie de la configuration de vos fonctions avant d'apporter des modifications. Il n'existe pas de gestion des versions ou de restauration intégrées pour les fonctions.

Supprimer une fonction

Détacher une fonction d'une configuration de lecture

Avant de supprimer une fonction, supprimez-la de toutes les configurations de lecture qui y font référence.

1. Ouvrez la MediaTailor console.
2. Dans le volet de navigation, sélectionnez Configurations.
3. Choisissez la configuration de lecture.
4. Dans la section Cartographie des fonctions, choisissez Modifier.
5. Supprimez la fonction du hook du cycle de vie.

6. Choisissez Enregistrer.

Suppression d'une fonction

Accédez à Fonctions, sélectionnez la fonction, puis choisissez Supprimer.

Règles de blocage des suppressions

MediaTailor vous empêche de supprimer une fonction toujours en cours d'utilisation.

Condition	Résultat
La fonction est associée à une configuration de lecture via le mappage des fonctions	La suppression est bloquée. Détachez d'abord la fonction.
La fonction est référencée dans un SEQUENTIAL_EXECUTOR	La suppression est bloquée. Supprimez d'abord la référence de la fonction parent.
La fonction n'est référencée nulle part	La suppression réussit.

Règles de validation

MediaTailor valide votre fonction lorsque vous la créez ou la mettez à jour. Les contrôles suivants sont effectués :

- Syntaxe des expressions — Toutes les expressions doivent être des JSonata valides.
- Fonctions restreintes — Les expressions ne peuvent pas appeler des fonctions JSonata restreintes. Consultez [Référence d'expression JSonata](#) la liste complète.
- Préfixes de clé de sortie : toutes les clés de sortie doivent commencer par un préfixe d'espace de noms reconnu. Pour la liste des préfixes acceptés, voir [Hooks de cycle de vie](#).
- Références de fonctions : tous les identifiants de fonction d'un FunctionList doivent faire référence à des fonctions existantes.
- Références circulaires : une fonction ne peut pas se référencer elle-même, directement ou indirectement.
- Profondeur d'imbrication — A SEQUENTIAL_EXECUTOR peut appeler d'autres fonctions, mais ces fonctions ne peuvent pas elles-mêmes être SEQUENTIAL_EXECUTOR s.

Pour les valeurs spécifiques et les limites de taille, voir [Restrictions](#).

Référence des API

Pour gérer les fonctions par programmation, consultez la référence des [AWS Elemental MediaTailor API](#).

Référence d'expression JSonata pour les fonctions

Cette page est une référence complète pour la syntaxe des expressions, les opérateurs et les fonctions disponibles dans Functions. Utilisez-le lorsque vous écrivez des expressions pour des blocs de sortie, des champs d'URL, des valeurs d'en-tête, des modèles de corps et des conditions d'exécution.

Délimiteurs d'expressions

Chaque valeur que vous définissez dans une fonction est une constante ou une expression, et non un mélange des deux. MediaTailor fait la distinction entre les deux à l'aide de délimiteurs.

Syntaxe	Type	Évaluation
<code>https://ads.example.com/vast</code>	Constant	Renvoyé tel quel sans évaluation.
<code>{%session.client_ip%}</code>	Expression	Évalué au moment de l'exécution. Le résultat remplace la valeur entière.
<code>GET</code>	Constant	Renvoyé tel quel.
<code>{%'https://ads.example.com/vast?ip=' & session.client_ip%'}</code>	Expression	Évalué au moment de l'exécution.

Important

Une valeur est soit entièrement une constante, soit entièrement une expression. Vous ne pouvez pas mélanger les deux en une seule valeur. Par exemple, `hello {% 'world' %}`

n'est pas valide. Pour associer du texte statique à des valeurs dynamiques, utilisez la concaténation de chaînes dans l'expression : `{%'hello ' & 'world'%}`

Bases de la langue

Notation par points pour la navigation sur le chemin

Utilisez la notation par points pour parcourir les données d'entrée. Chaque point descend d'un niveau dans la hiérarchie des objets.

```
session.client_ip           # the viewer's IP address
response.body.envelope      # a field inside a parsed JSON response
player_params.campaign_id   # a player parameter
```

Les champs manquants sont null renvoyés sans générer d'erreur :

```
temp.nonExistent           # null
temp.nonExistent.deeply.nested # null
```

Concaténation de chaînes avec **&**

L'opérateur joint deux valeurs de chaîne. Non-string les valeurs sont automatiquement converties en chaînes.

```
'https://ads.example.com/vast?ip=' & session.client_ip
# "https://ads.example.com/vast?ip=192.0.2.1"

'duration=' & 30
# "duration=30"
```

Expressions conditionnelles (ternaires)

Utilisez l'opérateur ternaire pour renvoyer l'une des deux valeurs en fonction d'une condition.

```
condition ? value_if_true : value_if_false
```

Exemples :

```
$exists(player_params.env) ? player_params.env : 'prod'  
response.statusCode = 200 ? response.body.id : 'unknown'  
$random() > 0.5 ? 'groupA' : 'groupB'
```

Vous pouvez imbriquer des expressions ternaires pour le branchement multidirectionnel :

```
$contains(session.user_agent, 'CTV') ? 'ctv'  
  : $contains(session.user_agent, 'Mobile') ? 'mobile'  
  : 'desktop'
```

Liaison de variables avec : =

Utilisez l':=opérateur entre parenthèses pour attribuer des valeurs intermédiaires dans une expression. Les variables liées sont limitées aux parenthèses qui les entourent et ne persistent pas en dehors de l'expression.

```
(  
  $base := 'https://ads.example.com';  
  $base & '/vast?ip=' & session.client_ip  
)
```

Des points-virgules séparent les instructions entre parenthèses. La dernière instruction est la valeur renvoyée par l'expression.

```
(  
  $code := response.statusCode;  
  $code != null and $code >= 200 and $code < 300  
    ? response.body.value  
    : 'fallback'  
)
```

Opérateurs

Arithmétique

Opérateur	Description	Exemple	Résultat
+	Addition	5 + 3	8

Opérateur	Description	Exemple	Résultat
-	Soustraction	10 - 4	6
*	Multiplication	6 * 7	42
/	Division	15 / 4	3.75
%	Modulo	17 % 5	2

Important

Les valeurs d'entrée provenant des paramètres du joueur et des données de session arrivent sous forme de chaînes. Utilisez-le `$number()` pour les convertir avant les comparaisons numériques ou arithmétiques. La comparaison d'une chaîne à un nombre produit des résultats inattendus.

Comparison (Comparaison)

Opérateur	Description	Exemple	Résultat
=	Egal à	<code>response. statusCode = 200</code>	true
!=	Non égal à	<code>player_pa rams.region != 'us-east-1'</code>	true si ce n'est pas us-east-1
<	Inférieur à	<code>avail.index < 3</code>	true si inférieur à 3
>	Supérieur à	<code>\$number(p layer_par ams.age) > 18</code>	true si vous avez plus de 18 ans
<=	Inférieur ou égal à	<code>\$count(items) <= 10</code>	true si 10 ou moins

Opérateur	Description	Exemple	Résultat
<code>>=</code>	Supérieur ou égal à	<code>response. statusCode >= 400</code>	true en cas d'état d'erreur

Booléen

Opérateur	Description	Exemple
<code>and</code>	ET logique	<code>response.statusCod e = 200 and \$exists(r esponse.body.id)</code>
<code>or</code>	OU logique	<code>player_params.regi on = 'us-east-1' or player_params.region = 'us-west-2'</code>

Utilisez des parenthèses pour la priorité :

```
score > 0.5 and (tier = 'premium' or tier = 'gold')
```

Note

À utiliser `$not()` pour une négation logique. Il n'existe aucun opérateur de not mot clé.

Adhésion (**en**)

L'`in`opérateur teste si une valeur existe dans un tableau.

```
'premium' in segments      # true if segments contains 'premium'  
player_params.region in ['us-east-1', 'us-west-2'] # true
```

Enchaînement (~ >)

L'opérateur de chaînage transmet le résultat de l'expression de gauche comme premier argument à la fonction de droite.

```
session.user_agent ~> $lowercase ~> $trim  
# equivalent to $trim($lowercase(session.user_agent))
```

Fonctions autorisées

MediaTailor prend en charge les fonctions intégrées suivantes. Toute fonction non répertoriée ici est bloquée et provoque une erreur de validation lorsque vous créez ou mettez à jour une fonction.

Conversion de type (3)

Fonction	Description	Exemple	Résultat
<code>\$string(value)</code>	Convertir en chaîne	<code>\$string(200)</code>	"200"
<code>\$number(value)</code>	Convertir en nombre	<code>\$number('42')</code>	42
<code>\$boolean(value)</code>	Convertir en booléen	<code>\$boolean(1)</code>	true

Introspection (4)

Fonction	Description	Exemple	Résultat
<code>\$length(string)</code>	String length	<code>\$length('hello')</code>	5
<code>\$count(array)</code>	Nombre d'éléments du tableau	<code>\$count([1, 2, 3])</code>	3
<code>\$exists(value)</code>	Vérifiez si la valeur existe (non indéfinie)	<code>\$exists(temp.id)</code>	true ou false
<code>\$keys(object)</code>	Obtenir les noms des clés des objets	<code>\$keys(response.body)</code>	["id", "name"]

Numérique (7)

Fonction	Description	Exemple	Résultat
<code>\$sum(array)</code>	Somme du tableau	<code>\$sum([1, 2, 3])</code>	6
<code>\$max(array)</code>	Valeur maximale	<code>\$max([10, 5, 20])</code>	20
<code>\$min(array)</code>	Valeur minimale	<code>\$min([10, 5, 20])</code>	5
<code>\$average(array)</code>	Moyenne arithmétique	<code>\$average([10, 20, 30])</code>	20
<code>\$abs(number)</code>	Valeur absolue	<code>\$abs(-7)</code>	7
<code>\$floor(number)</code>	Arrondir vers le bas	<code>\$floor(3.9)</code>	3
<code>\$round(number, precision)</code>	Du rond à la précision	<code>\$round(3.456, 2)</code>	3.46

Corde (7)

Fonction	Description	Exemple	Résultat
<code>\$uppercase(string)</code>	Vers les majuscules	<code>\$uppercase('hello')</code>	"HELLO"
<code>\$lowercase(string)</code>	Vers les minuscules	<code>\$lowercase('Hello')</code>	"hello"
<code>\$trim(string)</code>	Supprimer les espaces leading/trailing blancs	<code>\$trim(' hi ')</code>	"hi"

Fonction	Description	Exemple	Résultat
<code>\$substring(string, start, length)</code>	Extraire une sous-chaîne (base zéro)	<code>\$substring('abcdef', 2, 3)</code>	"cde"
<code>\$contains(string, pattern)</code>	Vérifiez si la chaîne contient un motif	<code>\$contains(session.user_agent, 'CTV')</code>	true ou false
<code>\$match(string, pattern)</code>	Faire correspondre la chaîne au modèle regex	<code>\$match('abc-123', /[0-9]+/)</code>	<code>{"match": "123", ...}</code>
<code>\$replace(string, pattern, replacement)</code>	Remplacer le modèle correspondant	<code>\$replace('hello', 'l', 'r')</code>	"herro"

Réseau (5)

Fonction	Description	Exemple	Résultat
<code>\$append(arr1, arr2)</code>	Concaténation de tableaux	<code>\$append([1, 2], [3, 4])</code>	[1, 2, 3, 4]
<code>\$reverse(array)</code>	Ordre inversé	<code>\$reverse([1, 2, 3])</code>	[3, 2, 1]
<code>\$sort(array)</code>	Tableau de tri	<code>\$sort([3, 1, 2])</code>	[1, 2, 3]
<code>\$distinct(array)</code>	Suppression des doublons	<code>\$distinct([1, 2, 2, 3])</code>	[1, 2, 3]
<code>\$map(array, func)</code>	Appliquer une fonction à chaque élément	<code>\$map([1,2,3], function(\$v) {\$v*2})</code>	[2, 4, 6]

Booléen (1)

Fonction	Description	Exemple	Résultat
<code>\$not(value)</code>	Logique : NON	<code>\$not(false)</code>	true

Aléatoire (1)

Fonction	Description	Exemple	Résultat
<code>\$random()</code>	Nombre aléatoire compris entre 0 (inclus) et 1 (exclusif)	<code>\$random() > 0.5 ? 'A' : 'B'</code>	"A" ou "B"

 Note

`$random()` génère une nouvelle valeur à chaque évaluation. Si vous avez besoin de la même valeur aléatoire dans plusieurs clés de sortie, liez-la d'abord à une variable : (`$r := $random(); ...`).

Date/time (4)

Fonction	Description	Exemple	Résultat
<code>\$now()</code>	Horodatage actuel sous forme de chaîne ISO 8601	<code>\$now()</code>	"2024-01-15T12:00:00.000Z"
<code>\$millis()</code>	Horodatage actuel en millisecondes depuis Epoch	<code>\$millis()</code>	1705320000000
<code>\$toMillis(string)</code>	Convertir une chaîne ISO 8601 en millisecondes	<code>\$toMillis('2024-01-15T12:00:00.000Z')</code>	1705320000000

Fonction	Description	Exemple	Résultat
		-15T12:00:00.000Z')	
\$fromMillis(number)	Convertir les millisecondes en chaîne ISO 8601	\$fromMillis(1705320000000)	"2024-01-15T12:00:00.000Z"

Encodage (6)

Fonction	Description	Exemple
\$encodeURIComponent(string)	Encodage d'URL (préserve les caractères structurels tels que/,?,&)	\$encodeURIComponent('https://example.com/path?q=hello world')
\$encodeURIComponentComponent(string)	Encode par URL un seul composant (code tous les caractères spéciaux)	\$encodeURIComponent('a&b=c') → "a%26b%3Dc"
\$decodeUrl(string)	Décoder une chaîne URL-encoded	\$decodeUrl('hello%20world') → "hello world"
\$decodeUrlComponent(string)	Décoder un composant URL-encoded	\$decodeUrlComponent('a%26b') → "a&b"
\$base64encode(string)	Encoder en Base64	\$base64encode('hello') → "aGVsbG8="
\$base64decode(string)	Décoder depuis Base64	\$base64decode('aGVsbG8=') → "hello"

 Tip

À utiliser `$encodeURIComponent()` pour les valeurs de paramètres de requête individuels.
À utiliser `$encodeURIComponent()` uniquement lorsque vous devez encoder une URL complète tout en préservant sa structure.

Schémas courants

Valeurs de repli

Indiquez une valeur par défaut lorsqu'une valeur n'existe peut-être pas.

```
{%$exists(player_params.region) ? player_params.region : 'us-east-1'%}
```

Construction d'URL dynamique

Créez une URL de serveur de décision publicitaire à partir de plusieurs entrées.

```
{%'https://ads.example.com/v1/vast?ip=' & $encodeURIComponent(session.client_ip) & 'ua=' & $encodeURIComponent(session.user_agent) & '&sid=' & session.id%}
```

Vérifications du code d'état pour la sortie HTTP_REQUEST

Protégez les valeurs de sortie contre les défaillances HTTP.

```
{%response.statusCode != null and response.statusCode = 200 ? response.body.envelope : 'default-envelope'%}
```

Conversion numérique à partir des paramètres du joueur

Les paramètres du joueur arrivent sous forme de chaînes. Convertissez-les avant les comparaisons arithmétiques ou numériques.

```
{%$number(player_params.max_duration) > 30 ? 'long' : 'short'%}
```

 Important

S'il `$number()` reçoit une chaîne non numérique, il renvoie `undefined`. Combinez-le avec le `$exists()` cas où le paramètre est peut-être manquant ou non valide : (`$val :=`

```
$number(player_params.max_duration); $exists($val) and $val > 30 ?  
'long' : 'short').
```

Répartition aléatoire du trafic

Assignez des spectateurs à des groupes d'expériences en utilisant `$random()`.

```
{% $random() > 0.5 ? 'https://ads.example.com/v1/vast-a' : 'https://ads.example.com/v1/  
vast-b'%}
```

Classification des types d'appareils

Classez les appareils en fonction de la chaîne de l'agent utilisateur.

```
{% $contains(session.user_agent, 'CTV') ? 'ctv' : $contains(session.user_agent,  
'Mobile') ? 'mobile' : 'desktop'%}
```

Exemples de fonctions

Avant de passer en revue ces exemples, vous devez connaître :

- [Hooks de cycle de vie](#)— Les champs d'entrée et les espaces de noms de sortie sont disponibles à chaque crochet du cycle de vie.
- [Types de fonctions et composition](#)— Le fonctionnement de chaque type de fonction et ses champs de configuration.
- [Création et gestion](#)— Comment créer des fonctions et les associer à des configurations de lecture.

Cette page fournit des configurations de fonctions complètes et fonctionnelles pour les cas d'utilisation courants. Chaque exemple inclut le scénario, la configuration complète, le mappage des fonctions et une explication de ce qui se passe lors de l'exécution de la fonction.

Exemple	Description du scénario
the section called “Exemple 1 : Enrichissement des données”	Récupérez une enveloppe LiveRamp d'identité au début de la session et stockez-la dans les paramètres du joueur.

Exemple	Description du scénario
the section called “Exemple 2 : répartition A/B du trafic”	Répartissez le trafic de demandes publicitaires de manière égale entre deux URL de serveur de décision publicitaire à des fins de A/B test.

Exemple 1 : Enrichissement des données

Scénario

Un service de streaming souhaite récupérer les données d'identité de l'audience auprès d'un service externe (dans ce cas, LiveRamp) au début de la session et les stocker dans les paramètres du lecteur pour le ciblage publicitaire plus tard dans la session. Une enveloppe d'identité est un identifiant crypté fourni par un service de résolution d'identité qui permet aux systèmes publicitaires de reconnaître un internaute sur tous les appareils et plateformes sans révéler d'informations personnelles.

Configuration

Récupérez LiveRamp l'enveloppe (HTTP_REQUEST) :

```
{
  "FunctionId": "fetchLiveRamp",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://api.example-identity.com/v2/envelope?pid=12345&iv=' &
player_params.hashId%}'",
    "Headers": {
      "Authorization": "{%'Bearer ramp-token-789'%}"
    },
    "RequestTimeoutMilliseconds": 2000,
    "Output": {
      "player_params.envelope_id": "{%response.body.envelopes[0].value%}"
    }
  }
}
```

Note

Bearer ramp-token-789 Remplacez-les par vos propres identifiants d'API. Ne stockez pas les jetons sensibles directement dans les configurations fonctionnelles en production. Envisagez d'utiliser une stratégie de rotation des jetons.

Définissez `RequestTimeoutMilliseconds` en fonction du temps de réponse attendu de votre service externe. Une valeur inférieure réduit la latence mais augmente le risque d'erreurs de temporisation.

Cartographie des fonctions

```
{
  "FunctionMapping": {
    "PRE_SESSION_INITIALIZATION": "fetchLiveRamp"
  }
}
```

Que se passe-t-il lorsque la fonction s'exécute

1. Un spectateur démarre une session de lecture.
2. MediaTailor exécute le hook `PRE_SESSION_INITIALIZATION` du cycle de vie et s'exécute `fetchLiveRamp`.
3. La fonction crée l'URL de demande à l'aide de l' `LiveRampAPI` `player_params.hashedException` et l'appelle. Dans l'URL, `pid` il y a l'ID du partenaire et `iv` l'identifiant haché du spectateur.
4. L'expression de sortie extrait la valeur de la première enveloppe du tableau de réponses (`response.body.envelopes[0].value`) et l'écrit dans `player_params.envelope_id`.

Tip

Pour gérer les erreurs de manière explicite, vérifiez `response.statusCode` avant d'accéder aux données de réponse : `{%response.statusCode = 200 ? response.body.envelopes[0].value : ''}`

Pour plus d'informations, consultez [Requête HTTP](#), [Pre-session initialisation](#), [Résolution des problèmes et surveillance](#) et [Restrictions](#).

Exemple 2 : répartition A/B du trafic

Scénario

Un service de streaming souhaite utiliser une fonction CUSTOM_OUTPUT pour répartir de manière aléatoire le trafic de demandes publicitaires entre deux URL de serveur de décision publicitaire (ADS) à des fins de test. A/B La moitié des demandes publicitaires sont destinées au point de terminaison v1 et l'autre moitié au point de terminaison v2.

Configuration

A/B répartition du trafic (CUSTOM_OUTPUT) :

```
{
  "FunctionId": "abTestAdsUrl",
  "FunctionType": "CUSTOM_OUTPUT",
  "CustomOutputConfiguration": {
    "Runtime": "JSONATA",
    "Output": {
      "adsRequest.url": "{%$random() < 0.5 ? 'https://ads.example.com/v1/decision?session=' & session.id : 'https://ads.example.com/v2/decision?session=' & session.id%}"
    }
  }
}
```

Dans cette configuration :

- **FunctionId**— Nom unique pour cette fonction.
- **FunctionType**— CUSTOM_OUTPUT évalue les expressions sans effectuer d'appels HTTP.
- **Output**— Correspond `adsRequest.url` à une expression qui sélectionne de manière aléatoire l'une des deux URL ADS.

Cartographie des fonctions

```
{
```



```
"FunctionMapping": {  
    "PRE_ADS_REQUEST": "abTestAdsUrl"  
}  
}
```

Que se passe-t-il lorsque la fonction s'exécute

1. Une interruption publicitaire se produit lors de la diffusion.
2. MediaTailor exécute le hook `PRE_ADS_REQUEST` du cycle de vie et s'exécute `abTestAdsUrl`.
3. La `$random()` fonction renvoie une valeur comprise entre 0 et 1. Si la valeur est inférieure à 0,5, la fonction définit l'URL ADS sur le point de terminaison v1. Sinon, il définit l'URL du point de terminaison v2.

La `$random()` fonction est évaluée à chaque pause publicitaire, de sorte que la répartition se fait par demande publicitaire, et non par session.

Tip

Pour ajuster le ratio de répartition du trafic, modifiez la valeur du seuil. Par exemple, `$random() < 0.8` envoie 80 % du trafic vers la première URL et 20 % vers la seconde.

Note

L'écriture vers `adsRequest.url` remplace l'URL ADS par défaut configurée dans la configuration de diffusion pour la pause publicitaire en cours.

Pour plus d'informations, consultez [Sortie personnalisée](#), [Pre-ads demande](#) et [Référence d'expression JSonata](#).

Fonctions de dépannage et de surveillance

Cette page vous aide à diagnostiquer les erreurs de fonctionnement courantes et à surveiller les performances des fonctions en production. La section de résolution des problèmes est organisée par symptôme : commencez par ce que vous observez, puis suivez la cause et corrigez le problème.

Contrôle

CloudWatch métriques

MediaTailor publie des métriques relatives à l'exécution des fonctions sur Amazon CloudWatch. Aucun opt-in n'est requis.

Hook-level métriques : un point de données par cycle de vie exécuté par un hook :

Métrique	Description	Dimensions
PreSessionInitHook. .Invocations	Nombre d'exécutions de hameçons	ConfigurationName
PreSessionInitHook. .Errors	Nombre d'erreurs de crochet	ConfigurationName
PreSessionInitHook. .Latency	Temps d'exécution du hook (ms)	ConfigurationName
PreAdsRequestHook. .Invocations	Nombre d'exécutions de hameçons	ConfigurationName
PreAdsRequestHook. .Errors	Nombre d'erreurs de crochet	ConfigurationName
PreAdsRequestHook. .Latency	Temps d'exécution du hook (ms)	ConfigurationName

Function-level métriques : un point de données par exécution de fonction individuelle :

Métrique	Description	Dimensions
Function. .Invocations	Nombre d'exécutions de fonctions	ConfigurationName , FunctionId , FunctionT ype , HookType

Métrique	Description	Dimensions
<code>Function.Errors</code>	Nombre d'erreurs de fonctionnement	<code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>
<code>Function.Latency</code>	Temps d'exécution de la fonction (ms)	<code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>

Pour plus de détails sur la configuration des alarmes et l'utilisation de ces métriques, consultez [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#).

Événements de journaux

MediaTailor émet des événements de journal pour l'exécution de la fonction. Les événements d'erreur sont émis par défaut. Les événements terminés et récapitulatifs sont facultatifs.

Type d'événement	Par défaut/ Opt-in	Groupe de journaux	Description
<code>PRE_SESSION_INIT_HOOK_SUMMARY</code>	Opt-in	Journal du manifeste	Résumé de l'exécution du hook (success/error)
<code>PRE_SESSION_INIT_HOOK_ERROR</code>	Par défaut	Journal du manifeste	Défaillance du hook avec <code>ErrorType</code> et cause
<code>PRE_SESSION_INIT_FUNCTION_COMPLETED</code>	Opt-in	Journal du manifeste	Fonction individuelle complétée par input/output
<code>PRE_SESSION_INIT_FUNCTION_ERROR</code>	Par défaut	Journal du manifeste	Défaillance fonctionnelle individuelle

Type d'événement	Par défaut/ Opt-in	Groupe de journaux	Description
PRE_ADS_REQUEST_HOOK_SUMMARY	Opt-in	Journal des interactions ADS	Résumé de l'exécution du hook (success/error)
PRE_ADS_REQUEST_HOOK_ERROR	Par défaut	Journal des interactions ADS	Défaillance du hook avec ErrorType et cause
PRE_ADS_REQUEST_FUNCTION_COMPLETED	Opt-in	Journal des interactions ADS	Fonction individuelle complétée par input/output
PRE_ADS_REQUEST_FUNCTION_ERROR	Par défaut	Journal des interactions ADS	Défaillance fonctionnelle individuelle

Pour activer le journal des événements opt-in, voir [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#).

Utilisez le `eventId` champ pour corréler les événements au niveau du hook et au niveau de la fonction pour une même exécution.

Les requêtes Amazon CloudWatch Logs Insights suivantes filtrent les événements d'erreur fonctionnelle `eventId` afin de suivre une seule exécution :

```
fields @timestamp, eventType, functionId, errorType, cause
| filter eventId = "5dc6f040-0f72-4e8c-a64e-25eeef62708c"
| sort @timestamp asc
```

Résolution des problèmes

Lorsqu'une fonction échoue, MediaTailor enregistre un `errorType` champ dans l'événement d'erreur. Utilisez ce champ pour identifier la classe de défaillance :

Error type (Type d'erreur)	Description
SYNTAX_ERROR	L'expression n'a pas pu être compilée ou a rencontré une erreur de type d'exécution
RESOURCE_LIMIT_ERROR	L'expression a dépassé les limites de temps du processeur, de mémoire ou de profondeur de pile
RESTRICTION_ERROR	L'expression a utilisé une fonction bloquée ou la limite de taille de la charge utile en entrée a été dépassée
TIMEOUT_ERROR	L'exécution de la fonction a dépassé sa limite de temps
VALIDATION_ERROR	Le chemin de sortie cible un champ non inscriptible dans la portée du hook actuel
INTERNAL_ERROR	Défaillance de l'infrastructure non liée à la fonction

Les entrées sont organisées par symptôme et font référence à ces types d'erreur, le cas échéant.

L'expression renvoie null de manière inattendue

Symptôme : une valeur de sortie que vous vous attendiez à renseigner est `null` ou est absente des paramètres du joueur.

Causes possibles :

Cause	Comment identifier	Corriger
Le champ de saisie n'existe pas à ce crochet du cycle de vie.	Vous l'avez référencé <code>adsRequest.url</code> dans une <code>PRE_SESSION_INITIALIZATION</code> fonction. Les données de demande ADS ne	Déplacez la fonction vers le crochet <code>PRE_ADS_REQUEST</code> du cycle de vie ou utilisez un autre champ de saisie.

Cause	Comment identifier	Corriger
	sont pas disponibles au début de la session.	Consultez Hooks de cycle de vie .
Le champ de saisie est absent des données de session.	Vous l'avez référencé <code>player_params.campaign_id</code> , mais le joueur n'a pas transmis ce paramètre lors de l'initialisation de la session.	<code>\$exists()</code> À utiliser pour vérifier avant d'accéder : <code>{% \$exists(player_params.campaign_id) ? player_params.campaign_id : 'default' %}</code> .
Vous avez écrit un objet ou un tableau dans les paramètres du joueur ou dans la requête ADS.	Ces espaces de noms n'acceptent que des chaînes, des nombres et des booléens. Les objets et les tableaux sont filtrés.	Stockez des données complexes temp.* et extrayez-en des chaînes, des nombres ou des booléens lors d'une étape ultérieure.

RESOURCE_LIMIT_ERROR : Débordement de pile

Symptôme : La fonction échoue avec `errorType` : "RESOURCE_LIMIT_ERROR" et `cause` : "Stack overflow error".

Cause : L'expression a dépassé la profondeur de pile maximale de 100 niveaux. Cela se produit généralement avec des expressions conditionnelles (if/then/else) profondément imbriquées ou des affectations de variables complexes.

Cela signifie que l'expression comporte trop de niveaux d'imbrication MediaTailor pour être traitée.

Correctif : Simplifiez l'expression. Divisez une logique complexe en plusieurs entrées de sortie ou en plusieurs étapes dans un exécuteur séquentiel.

RESOURCE_LIMIT_ERROR : délai d'expiration du processeur

Symptôme : La fonction échoue avec `errorType` : "RESOURCE_LIMIT_ERROR" et `cause` : "Expression evaluation timeout: Check for infinite loop".

Cause : L'expression a dépassé la limite de temps du processeur de 100 ms. Cela peut se produire avec des expressions qui effectuent des calculs coûteux sur de grandes structures de données.

Correctif : réduisez la complexité de l'expression. Si vous traitez de grands tableaux, pensez à déplacer cette logique vers un service externe et à l'appeler avec une `HTTP_REQUEST` fonction.

RESTRICTION_ERROR : Fonction non autorisée

Symptôme : La fonction échoue avec `errorType`: "RESTRICTION_ERROR" et `cause`: "Function '<name>' is not allowed".

Cause : L'expression appelle une fonction JSonata qui ne figure pas dans la liste autorisée des 38 fonctions. Les exemples courants incluent `$filter`, `$reduce`, `$eval`, `$split`, et `$join`.

Correctif : Vérifiez `cause` dans le champ le nom de la fonction bloquée. Remplacez-le par une alternative autorisée. Consultez [Référence d'expression JSonata](#) la liste complète des 38 fonctions autorisées.

Les fonctions autorisées couramment utilisées incluent `$string`, `$number`, `$substring`, `$contains`, et `$encodeURIComponent`.

RESTRICTION_ERROR : Expression trop longue

Symptôme : La fonction ne parvient pas à être créée ou mise à jour avec `cause`: "Expression length <actual> exceeds limit <limit>".

Cause : une seule expression dépasse 1 000 caractères.

Correctif : Divisez l'expression en parties plus petites. Utilisez plusieurs entrées de sortie ou répartissez la logique en plusieurs étapes dans un exécuteur séquentiel. Utilisez la liaison variable (`:=`) pour éviter de répéter de longues sous-expressions.

Erreur HTTP : le code d'état est nul

Symptôme : Dans le résultat d'une `HTTP_REQUEST` fonction, `response.statusCode` c'est `null`.

Cause : L'API externe était inaccessible, le délai de connexion a expiré ou une erreur réseau s'est produite. Lorsque cela se produit, MediaTailor définit `response.body` sur `null`, `response.statusCode` to et `response.text` to "Internal Error".

Correctif : vérifiez toujours `response.statusCode` avant d'accéder aux données de réponse :

```
{%response.statusCode = 200 ? response.body.value : 'default'%}
```

Cette expression vérifie si l'appel HTTP a renvoyé un code d'état 200. Si c'est le cas, il utilise la valeur de réponse. Dans le cas contraire, la valeur par défaut est rétablie.

Si cela se produit fréquemment, vérifiez si l'API externe est saine. Envisagez de l'augmenter `RequestTimeoutMilliseconds` si l'API est lente ou de la diminuer si vous souhaitez échouer rapidement.

Erreur HTTP : le corps de la réponse est nul

Symptôme : 200 mais `response.statusCode` c'`response.body` est le cas `null`.

Cause : le corps de la réponse n'est pas un JSON valide ou dépasse 20 000 caractères. MediaTailor analyse uniquement les réponses JSON contenant jusqu'à 20 000 caractères. `response.body`

Correctif : à utiliser `response.text` comme solution de secours. Le `response.text` champ contient le corps de réponse brut tronqué à 20 000 caractères :

```
{%response.statusCode = 200 ? ($exists(response.body.id) ? response.body.id :  
  $substring(response.text, 0, 100)) : 'error'%}
```

Si les données dont vous avez besoin dépassent la limite de 20 000 caractères, pensez à demander à l'API externe de renvoyer une réponse plus petite (par exemple, en demandant des champs spécifiques).

Erreur HTTP : échec de la validation de l'URL

Symptôme : La fonction échoue lors de l'exécution avec un message indiquant que l'URL est mal formée, utilise un schéma non valide ou dépasse la longueur maximale.

Causes possibles :

Cause	Corriger
L'URL n'utilise pas <code>http</code> ou <code>https</code> .	Assurez-vous que l'expression d'URL produit une URL commençant par <code>http://</code> ou <code>https://</code> .
L'URL dépasse 2 048 caractères après évaluation de l'expression.	Raccourcissez l'URL. Déplacez de grandes valeurs de paramètres vers le corps de la requête à l'aide d'une méthode POST.
L'URL est mal formée (il ne s'agit pas d'un URI valide).	Vérifiez la présence de caractères manquants ou supplémentaires dans l'expression.

Cause	Corriger
	<code>\$encodeURIComponent()</code> À utiliser pour les valeurs des paramètres de requête susceptibles de contenir des caractères spéciaux.

ERREUR_DE_VALIDATION

Symptôme : La fonction échoue avec `errorType` : "VALIDATION_ERROR". Cette erreur peut se produire au moment de la création (lorsque vous créez ou mettez à jour la fonction) ou au moment de l'exécution (lorsque la fonction s'exécute pendant une session).

Causes possibles :

Cause	Exemple	Corriger
La clé de sortie cible un espace de noms non accessible en écriture sur le hook actuel.	Écrire à <code>adsRequest.url</code> dans une fonction attachée à <code>PRE_SESSION_INITIALIZATION</code> .	Vérifiez quels espaces de noms de sortie sont autorisés lors de votre cycle de vie. <code>PRE_SESSION_INITIALIZATION</code> autorise uniquement <code>player_params.*</code> . Déplacez la fonction sur le crochet approprié ou modifiez la clé de sortie. Consultez Hooks de cycle de vie .
La clé de sortie utilise des caractères non valides.	Une clé de sortie comme <code>player_params.device type</code> (avec un espace). Seuls les lettres, les chiffres, les traits de soulignement et les traits d'union sont autorisés.	Renommez la clé de sortie pour n'utiliser que des caractères valides. Par exemple, utilisez <code>player_params.device_type</code> plutôt.
La clé de sortie ne commence pas par un préfixe valide.	Une clé de sortie comme <code>custom.myValue</code> au lieu	Utilisez un préfixe de sortie valide : <code>player_pa</code>

Cause	Exemple	Corriger
	de <code>player_params.myValue</code> ou <code>temp.myValue</code> .	<code>rams.* temp.*</code> , <code>ouadsRequest.*</code> .
Une expression JSonata contient une erreur de syntaxe.	Un devis de clôture manquant ou une condition incomplète : <code>{%session.id & %}</code> .	Vérifiez l'expression pour détecter les guillemets manquants, les parenthèses non correspondantes ou les opérateurs non pris en charge tels que <code>ou. ?? ? :</code>
Il manque un champ obligatoire dans une fonction <code>HTTP_REQUEST</code> .	Le champ URL est vide ou la méthode n'est pas spécifiée.	Assurez-vous que les champs URL et méthode sont définis. La méthode doit être GET ou POST.
L'URL créée par l'expression n'est pas valide.	L'URL évaluée utilise un schéma non pris en charge, par exemple <code>ftp://</code> , dépasse 2 048 caractères ou est mal formée.	Vérifiez que l'expression d'URL produit une <code>https://</code> URL valide <code>http://</code> . <code>\$encodeURIComponent()</code> À utiliser pour les valeurs des paramètres de requête susceptibles de contenir des caractères spéciaux.
Un en-tête HTTP contient des caractères non valides ou utilise un nom restreint.	Une valeur d'en-tête contient des sauts de ligne, ou le nom de l'en-tête est <code>host</code> ou <code>transfer-encoding</code> .	Supprimez les caractères non valides des valeurs d'en-tête . Évitez les noms d'en-tête restreints. Voir Requête HTTP pour les limites d'en-têtes.

Vérifiez le champ cause dans le journal des erreurs : il identifie le champ ou l'expression qui a échoué à la validation.

INTERNAL_ERROR

Symptôme : La fonction échoue avec `errorType` : "INTERNAL_ERROR".

Cause : Une défaillance de l'infrastructure n'est pas liée à la configuration de vos fonctions.

Correctif : Réessayez la demande. Si l'erreur persiste, contactez le AWS Support.

Limites des fonctions

Cette page répertorie les limites qui s'appliquent aux fonctions. Ces limites sont appliquées au moment de la création (lorsque vous créez ou mettez à jour une fonction) et au moment de l'exécution (lorsque la fonction s'exécute).

Limites d'exécution des crochets

Crochet	Limite	Value
PRE_SESSION_INITIALIZATION	Délai d'attente total	2 000 ms
PRE_SESSION_INITIALIZATION	player_params Taille totale	1 000 caractères
PRE_ADS_REQUEST	Délai d'attente total	2 000 ms

Ces délais couvrent l'intégralité de l'exécution du hook, y compris toutes les étapes de la fonction et les appels HTTP. Si le hook dépasse son délai d'expiration, MediaTailor supprime toutes les sorties et procède comme si aucune fonction n'était attachée.

Les délais d'expiration des fonctions individuelles (comme `RequestTimeoutMilliseconds` pour les fonctions `HTTP_REQUEST`) doivent correspondre au délai d'expiration total du hook. Par exemple, si le délai d'expiration du hook est de 2 000 ms et qu'une fonction `HTTP_REQUEST` est définie `RequestTimeoutMilliseconds` sur 2 000 ms, la fonction peut expirer avant de se terminer si un traitement est effectué avant ou après l'appel HTTP.

La taille `player_params` totale inclut le nombre de caractères combiné de tous les noms et valeurs des clés de sortie. Par exemple, la sortie `player_params.deviceType = "mobile"` et le nombre de `player_params.region = "us-east-1"` caractères `deviceType mobileregion`, et par rapport au total `us-east-1` de 1 000 caractères.

Limites d'expression

Limite	Value	Appliqué à
Temps CPU par expression	100 ms	Environnement d'exécution
Profondeur de la pile	100	Environnement d'exécution
Longueur de l'expression	1 000 caractères	Conception

Lorsqu'une expression dépasse une limite lors de l'exécution, MediaTailor arrête l'expression et enregistre l'erreur. Pour plus de détails sur les types d'erreurs spécifiques, consultez [Résolution des problèmes et surveillance](#).

Limites de composition fonctionnelle

Limite	Value
Étapes par exécuteur séquentiel	1-10
Enfants par exécuteur testamentaire simultané	1-10
Simultanéité maximale (exécuteur simultané)	1—2
Délai d'attente par défaut (exécuteur simultané)	2 000 ms
Espaces de noms uniques par exécuteur simultané	Obligatoire (validé au moment de la rédaction)
Profondeur de nidification maximale	2
Nombre total d'exécutions de fonctions par hook	20
Entrées de sortie par fonction	20
Références circulaires	Non autorisée

Ces limites sont appliquées au moment de la création.

Limites de requêtes HTTP

Limite	Value
Longueur de l'URL	2 048 caractères
Demander la taille du corps	64 Ko
Nombre d'en-têtes	50
Longueur du nom d'en-tête	256 caractères
Longueur de la valeur d'en-tête	8,192 caractères
Délai d'expiration de la demande	100 à 2 000 ms (configurable par le client via) <code>RequestTimeoutMilliseconds</code>
Schémas d'URL autorisés	<code>https</code> , <code>http</code>
En-têtes restreints	<code>Host</code> , <code>Transfer-Encoding</code> , <code>Content-Length</code> , <code>Connection</code>

La longueur de l'URL, la taille du corps, le nombre d'en-têtes et les limites de taille d'en-tête sont appliqués au moment de la création. Le délai d'expiration des requêtes et les en-têtes restreints sont appliqués lors de l'exécution.

Si une fonction définit un en-tête restreint, elle MediaTailor accepte la configuration de la fonction au moment de la création mais supprime l'en-tête lorsque la requête HTTP est envoyée pendant l'exécution.

Limites de réponse

Limite	Value
<code>response.body</code> taille maximale	20 000 caractères
<code>response.text</code> longueur maximale	20 000 caractères

Ces limites sont appliquées au moment de l'exécution.

Limites de sortie

Limite	Value
Longueur maximale de la valeur de sortie (par touche)	1 000 caractères

Ces limites sont appliquées au moment de l'exécution.

Fonctions JSonata autorisées

Pour obtenir la liste complète des fonctions autorisées, consultez [Référence d'expression JSonata](#).

Pour obtenir de l'aide pour diagnostiquer les erreurs liées à ces limites, consultez [Résolution des problèmes et surveillance](#).

MediaTailor variables publicitaires dynamiques pour les demandes ADS

AWS Elemental MediaTailor utilise des variables publicitaires dynamiques pour transmettre les informations de votre session de visionnage au serveur de décision publicitaire (ADS). Ces informations permettent à l'ADS de sélectionner les publicités les plus pertinentes pour vos spectateurs.

Cette section fournit une vue d'ensemble des variables publicitaires dynamiques et des liens vers des guides de mise en œuvre spécifiques. Pour obtenir des instructions de configuration étape par étape, consultez les différentes rubriques ci-dessous.

Types de variables dynamiques

MediaTailor prend en charge quatre types de variables dynamiques :

- Variables de session : valeurs générées automatiquement, telles que l'ID de session et SCTE-35 les données. Consultez [MediaTailor variables de session pour les requêtes ADS](#).
- Variables du lecteur : paramètres personnalisés envoyés par votre lecteur vidéo. Consultez [MediaTailor variables de lecteur pour les requêtes ADS](#).

- Variables de domaine avec alias de configuration — Domaines URL dynamiques pour les configurations multi-origines.
- Variables de service : `aws.*` paramètres réservés qui contrôlent le comportement au niveau de la session dans MediaTailor. Consultez [Variables de service](#).
- Alias de configuration : mappages prédéfinis pour le remplacement dynamique des variables. Consultez [Alias de configuration](#).

Cas d'utilisation courants

Utilisez des variables publicitaires dynamiques pour :

- Transmettez les données démographiques et les préférences des spectateurs à votre ADS
- Demandes d'acheminement vers différentes origines en fonction de la situation géographique
- Permettre un affichage décalé dans le temps grâce à l'intégration MediaPackage
- Mettre en œuvre A/B des scénarios de test et de basculement

Les sections suivantes fournissent des informations supplémentaires sur l'utilisation de variables publicitaires dynamiques avec MediaTailor.

Rubriques

- [MediaTailor variables de session pour les requêtes ADS](#)
- [MediaTailor variables de lecteur pour les requêtes ADS](#)
- [MediaTailor variables de domaine pour plusieurs sources de contenu](#)
- [MediaTailor variables de service pour le contrôle de session](#)
- [MediaTailor vue d'ensemble des alias de configuration](#)
- [MediaTailor transmission de paramètres à ADS](#)
- [MediaTailor routage des paramètres pour ADS et origines](#)
- [MediaTailor et intégration de visualisation MediaPackage décalée dans le temps](#)
- [MediaTailor comportement et persistance des sessions de paramètres](#)
- [MediaTailor référence des paramètres et limites](#)
- [MediaTailor guide de dépannage des paramètres](#)
- [MediaTailor guide de dépannage des alias de configuration](#)

Pour connaître les exigences relatives au formatage des paramètres et la résolution des problèmes, reportez-vous [MediaTailor référence des paramètres et limites](#) aux sections et [MediaTailor guide de dépannage des paramètres](#).

MediaTailor variables de session pour les requêtes ADS

AWS Elemental MediaTailor envoie des données de session à l'Ad Decision Server (ADS) lorsque vous configurez AWS Elemental MediaTailor pour spécifier une ou plusieurs des variables répertoriées dans cette section dans le modèle d'URL ADS. Vous pouvez utiliser des variables individuelles et concaténer plusieurs variables pour créer une valeur unique. MediaTailor génère certaines valeurs et obtient le reste à partir de sources telles que le manifeste et la demande d'initialisation de session du joueur.

Le tableau suivant décrit les variables de données de session que vous pouvez utiliser dans la configuration de l'URL de votre modèle de demande ADS. Les numéros de section répertoriés dans le tableau correspondent à la version 2019a de la spécification S-35 de la Society of Cable Telecommunications Engineers (SCTE), [Digital Program Insertion Cueing Message](#). Pour plus de détails sur la prélecture des annonces, voir [Publicités de prélecture](#).

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[avail.in dex]	Oui		Nombre qui représente la position d'une annonce disponible dans un index. Au début d'une session de lecture, MediaTailor crée un index de toutes les publicités disponibles dans un manifeste et stocke l'index pour le reste de la session. Lorsqu'il MediaTailor fait une demande à l'ADS pour remplir l'offre, il inclut le numéro d'index de disponibilité des annonces. Ce paramètre permet au serveur ADS d'améliorer la sélection des publicités en utilisant des fonctionnalités telles que l'exclusion concurrentielle et le plafonnement des fréquences.

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[avail.random]	Oui		Un nombre aléatoire compris entre 0 et 10 000 000 000, sous forme de nombre long, MediaTailor généré pour chaque demande adressée à l'ADS. Certains serveurs publicitaires utilisent ce paramètre pour activer des fonctionnalités telles que la séparation de publicités d'entreprises concurrentes.
[scte.archive_allowed_flag]	Oui	10.3.3.1	Valeur booléenne facultative. Lorsque cette valeur est égale à 0, des restrictions d'enregistrement sont appliquées au segment. Lorsque cette valeur est égale à 1, aucune restriction d'enregistrement n'est imposée au segment.
[scte.avail_num]	Oui	9.7.2.1	La valeur analysée par MediaTailor le SCTE-35 champavail_num , sous la forme d'un nombre long. MediaTailor Je peux utiliser cette valeur pour désigner des numéros de disponibilité linéaires. La valeur doit être un entier.
[scte.avails_expected]	Oui	9,7,2.1	Une valeur longue facultative qui indique le nombre attendu de résultats au cours de l'événement en cours.
[scte.delivery_not_restricted_flag]	Oui	10.3.3.1	Valeur booléenne facultative. Lorsque cette valeur est égale à 0, les cinq bits suivants sont réservés. Lorsque cette valeur est égale à 1, les cinq bits suivants prennent la signification décrite dans la SCTE-35 spécification.

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[scte.device_restrictions]	Oui	10.3.3.1	Valeur entière facultative qui signale trois groupes de périphériques prédéfinis, indépendants et non hiérarchiques. Pour plus d'informations sur cette variable, consultez la description <code>segments_expected</code> dans la spécification. SCTE-35
[scte.event_id]	Oui	9.1 et 9.7.2.1	La valeur analysée par MediaTailor le SCTE-35 champs <code>splICE_event_id</code> , sous la forme d'un nombre long. MediaTailor utilise cette valeur pour désigner des numéros linéaires de disponibilité des annonces ou pour renseigner les chaînes de requête du serveur publicitaire, telles que la position des pods publicitaires. La valeur doit être un entier.
[scte.no_regional_blackout_flag]	Oui	10.3.3.1	Valeur booléenne facultative. Lorsque cette valeur est égale à 0, des restrictions d'interdiction régionale s'appliquent au segment. Lorsque cette valeur est égale à 1, les restrictions d'interdiction régionales ne s'appliquent pas au segment.
[scte.segment_num]	Oui	10.3.3.1	Valeur entière facultative qui numérote les segments d'une collection de segments. Pour plus d'informations sur cette variable, consultez la description <code>segment_num</code> dans la spécification. SCTE-35
[scte.segmentation_event_id]	Oui	10.3.3.1	MediaTailor expose cette variable sous la forme scte.event_id .

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[scte.segmentation_type_id]	Oui	10.3.3.1	Valeur entière 8 bits facultative qui spécifie le type de segmentation. Pour plus d'informations sur cette variable, consultez la description <code>segmentation_type_id</code> dans la spécification. SCTE-35

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[scte.seg mentation _upid]	segmentat ion_upid_ type : Oui private_d ata : Oui	segmentat ion_upid : 10.3.3.1 UPID privé géré : 10.3.3.3	Correspond à l' SCTE-35 segmentat ion_upid élément. L'segmentat ion_upid élément contient segmentation_upid_ type et segmentation_upid_length . MediaTailor prend en charge les segmentat ion_upid types suivants : - Informations sur l'ADS (0x0E) : informations publicitaires. Pour plus d'informations, consultez la description de segmentation_upid dans la spécifica tion. SCTE-35 - UPID privé géré (0x0C) : structure UPID privé géré (MPU) telle que définie dans la spécification. SCTE-35 MediaTailor prend en charge les représent ations SCTE binaires ou DASH XML. Vous pouvez utiliser cette structure dans un flux de travail de podbuster. Pour ce faire, spécifiez une valeur de 32 bits (4 octets) format_id entifier et incluez les paramètres suivants dans l'private_data attribut : <pre>ABCD{"assetId": " my_program ", "cueData": {"cueType": " theAdType ", "key": " pb", "value": " 123456"}}</pre> MediaTailor analyse les valeurs du JSON précédent et les transmet aux variables scte.segm entation_upid.assetId scte.segm entation_upid.cueData.key , et

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
			<code>scte.segmentation_upid.cueData.value</code> dynamiques. Défini par l'utilisateur (0x01) : structure définie par l'utilisateur. Pour plus d'informations, consultez la description de <code>segmentation_upid</code> dans la spécification. SCTE-35
[<code>scte.segmentation_upid.assetId</code>]	Oui		Utilisé conjointement avec l'UPID privé géré (0xC) <code>segmentation_upid_type</code> pour les flux de travail Podbust. MediaTailor dérive cette valeur à partir du <code>assetId</code> paramètre de la structure <code>private_data</code> JSON du MPU. Pour de plus amples informations, veuillez consulter Managed Private UPID JSON structure for a podbust workflow .
[<code>scte.segmentation_upid.cueData.key</code>]	Oui		Utilisé conjointement avec l'UPID privé géré (0xC) <code>segmentation_upid_type</code> pour les flux de travail Podbust. MediaTailor dérive cette valeur à partir du <code>cueData.key</code> paramètre de la structure <code>private_data</code> JSON du MPU. Pour de plus amples informations, veuillez consulter Managed Private UPID JSON structure for a podbust workflow .

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[scte.segmentation_upid.cueData.value]	Oui		Utilisé conjointement avec l'UPID privé géré (0xC) <code>segmentation_upid_type</code> pour les flux de travail Podbuster. MediaTailor dérive cette valeur à partir du <code>cueData.key</code> paramètre de la structure <code>private_data</code> JSON du MPU. Pour de plus amples informations, veuillez consulter Managed Private UPID JSON structure for a podbuster workflow. La valeur peut être une chaîne.
[scte.segmentation_upid.private_data.{index}]	Oui		Utilisé conjointement avec l'UPID privé géré (0xC) <code>segmentation_upid_type</code> pour les flux de travail publicitaires ciblés. MediaTailor divise les jetons UPID de segmentation séparés par des deux-points et crée des variables de session indexées. L'index correspond à la position dans la liste délimitée par deux points, sans tenir compte des espaces en tête des deux points initiaux. Par exemple, si <code>segmentation_upid = ":3213214:2313321/5:3943"</code>, alors : <code>[scte.segmentation_upid.private_data.0]</code> = 3213214 <code>[scte.segmentation_upid.private_data.1]</code> = 2313321/5 <code>[scte.segmentation_upid.private_data.2]</code> = 3943 La valeur peut être une chaîne.

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[scte.segments_expected]	Oui	10.3.3.1	Valeur entière facultative qui indique le nombre attendu de segments individuels au sein d'une collection de segments. Pour plus d'informations sur cette variable, consultez la description segments_expected dans la spécification. SCTE-35
[scte.sub_segment_num]	Oui	10.3.3.1	Valeur entière facultative qui identifie un sous-segment particulier au sein d'un ensemble de sous-segments. Pour plus d'informations sur cette variable, consultez la description de sub_segment_num dans la spécification. SCTE-35
[scte.sub_segments_expected]	Oui	10.3.3.1	Valeur entière facultative qui donne le nombre attendu de sous-segments individuels au sein d'une collection de sous-segments. Pour plus d'informations sur cette variable, consultez la description de sub_segments_expected dans la spécification. SCTE-35
[scte.unique_program_id]	Oui	9.7.2.1	La valeur entière analysée par MediaTailor le SCTE-35 splice_insert champ. unique_program_id L'ADS utilise l'ID de programme unique (UPID) pour fournir un ciblage publicitaire au niveau du programme pour les flux linéaires en direct. Si la SCTE-35 commande n'est pas splice insert, MediaTailor définissez-la sur une valeur vide. La valeur doit être un entier.

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[session.avail_duration_ms]	Oui		Durée en millisecondes du créneau de disponibilité des annonces. La valeur par défaut est de 300 000 ms. AWS Elemental MediaTailor obtient la valeur de durée à partir du manifeste d'entrée comme suit : • Pour HLS : MediaTailor obtient la durée à partir des valeurs #EXT-X-CUE-OUT: DURATION ou de la #EXT-X-DATERANGE balise. Si le manifeste d'entrée a une durée nulle, non valide ou nulle pour la disponibilité de l'annonce dans ces balises, MediaTailor utilise la valeur par défaut. • Pour DASH : MediaTailor obtient la valeur de durée à partir de la durée de l'événement, si celle-ci est spécifiée. Sinon, la valeur par défaut est utilisée. • Pour la VOD : lorsqu'un flux VOD déclenche un appel publicitaire pré-roll, si le manifeste n'inclut pas la messagerie SCTE avec une valeur de durée, aucune durée MediaTailor n'est saisie pour le [session.avail_duration_ms], y compris la valeur de durée par défaut.
[session.avail_duration_secs]	Oui		Durée en secondes du créneau de disponibilité des annonces, ou durée de disponibilité des annonces, arrondie à la seconde la plus proche. MediaTailor détermine cette valeur de la même manière qu'elle le détermine [session.avail_duration_ms] .
[session.client_ip]	Non		Adresse IP distante d'où provient la MediaTailor demande. Si l'en-tête X-forwarded-for est défini, cette valeur est ce que MediaTailor utilise pour client_ip .

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[session. id]	Non		Identifiant numérique unique pour la session de lecture en cours. Toutes les demandes adressées par un joueur pour une session ont le même ID et, par conséquent, il peut être utilisé pour les champs ADS destinés à établir une corrélation entre les demandes d'une même visualisation.
[session. referer]	Non		Généralement, l'URL de la page qui héberge le lecteur vidéo. MediaTailor définit cette variable sur la valeur de l'Referer en-tête que le joueur a utilisé dans sa requête MediaTailor. Si le lecteur ne fournit pas cet en-tête, MediaTailor laisse vide [session.referer] . Si vous utilisez un réseau de diffusion de contenu (CDN) ou un proxy devant le point de terminaison du manifeste et que vous souhaitez que cette variable apparaisse, utilisez le bon en-tête depuis le lecteur situé ici.
[session. user_agen t]	Non		User-Agent En-tête MediaTailor reçu suite à la demande d'initialisation de session du joueur. Si vous utilisez un réseau de diffusion de contenu (CDN) ou un proxy devant le point de terminaison du manifeste, utilisez comme proxy l'en-tête correct du lecteur ici.
[session. uuid]	Non		Alternative à [session.id] . Il s'agit d'un identifiant unique pour la session courante de lecture, tel que le suivant : e039fd39-09f0-46b2-aca9-9871cc116cde

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description
[avail.source_content_time_epoch_ms]	Non		Pour HLS, la valeur est le PDT du segment d'origine à l'origine de l'utilisation. Pour DASH, la valeur est <code><SupplementalProperty> urn:scte:dash:utc-time</code> celle <code><Period></code> qui contient le <code><EventStream></code> . • Pour les publicités HLS ou DASH avant le <code>[avail.source_content_time_epoch_ms]</code> lancement, il s'agit du PDT du premier segment HLS et du <code><SupplementalProperty> urn:scte:dash:utc-time</code> premier. <code><Period></code> Pour les diffusions en direct avec de courtes fenêtres de manifeste, cette valeur sera différente pour chaque spectateur en fonction du moment où il commence à regarder le flux. Pour une diffusion en direct avec une fenêtre DVR complète, la valeur sera la même pour chaque spectateur. • Pour les disponibilités partielles dues à une suppression de disponibilité, il <code>[avail.source_content_time_epoch_ms]</code> s'agit du PDT du segment de contenu source qui a déclenché la disponibilité. Par exemple, si MediaTailor les 20 premières secondes sont supprimées, le PDT d'origine <code>[avail.source_content_time_epoch_ms]</code> sera toujours réglé sur le PDT d'origine, et non décalé de 20 secondes. • Pour les requêtes ADS par prélecture, <code>[avail.source_content_time_epoch_ms]</code> est vide, mais les clients peuvent le définir à l'aide de

Nom	Disponibl e pour la prélectur e des annonces	SCTE-35 section des spécifica tions	Description

variables dynamiques de récupération du calendrier de prélecture.

Exemple

Si le serveur ADS nécessite un paramètre de requête nommé `deviceSession` pour le transmettre avec l'identificateur de session unique, le modèle d'URL du serveur ADS dans AWS Elemental MediaTailor peut se présenter comme suit :

```
https://my.ads.server.com/path?deviceSession=[session.id]
```

AWS Elemental MediaTailor génère automatiquement un identifiant unique pour chaque flux et saisit l'identifiant à la place `desession.id`. Si l'identifiant est `1234567`, la demande finale MediaTailor adressée à l'ADS ressemblera à ceci :

```
https://my.ads.server.com/path?deviceSession=1234567
```

Si l'ADS nécessite la transmission de plusieurs paramètres de requête, le modèle d'URL ADS AWS Elemental MediaTailor peut ressembler à ce qui suit :

```
https://my.ads.server.com/sample?  
e=[scte.avails_expected]&f=[scte.segment_num]&g=[scte.segments_expected]&h=[scte.sub_segment_num]
```

L'exemple de fragment XML de marqueur DASH suivant montre comment utiliser `scte35:SpliceInsert` :

```
<Period start="PT444806.040S" id="123456" duration="PT15.000S">  
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">  
    <Event duration="1350000">  
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832" tier="4095">  
        <scte35:SpliceInsert spliceEventId="1234567890"  
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"  
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
```

```

    <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></scte35:Program>
    <scte35:BreakDuration autoReturn="true" duration="1350000"/>
  </scte35:SpliceInsert>
</scte35:SpliceInfoSection>

```

L'exemple de fragment XML de marqueur DASH suivant montre comment utiliser `scte35:TimeSignal` :

```

<Period start="PT346530.250S" id="123456" duration="PT61.561S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="5310000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183003" tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="3442857000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="1234567"
segmentationEventCancelIndicator="false" segmentationDuration="8100000"
segmentationTypeId="52" segmentNum="0" segmentsExpected="0">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="false"
noRegionalBlackoutFlag="false" archiveAllowedFlag="false" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="12"
segmentationUpidLength="2">0100</scte35:SegmentationUpid>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
</Period>

```

L'exemple de fragment XML de marqueur DASH suivant montre comment utiliser `scte35:Binary` :

```

<Period start="PT444806.040S" id="123456" duration="PT15.000S">
  <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
    <Event presentationTime="1541436240" duration="24" id="29">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>DAhAAAAAAAAAAP/wEAUAAAHaf+9/fgAg9YDAAAAAAAAA25aoh</Binary>
      </scte35:Signal>
    </Event>
    <Event presentationTime="1541436360" duration="24" id="30">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmLuZyBmb3IgaWZlZjB2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYmluYXJ5Lg==
Binary>
      </scte35:Signal>
    </Event>
  </EventStream>
</Period>

```

L'exemple de balise HLS suivant montre comment utiliser EXT-X-DATERANGE :

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2014-03-05T11:
15:00Z",PLANNED-DURATION=59.993,SCTE35-OUT=0xFC002F0000000000FF0
00014056FFFFFFF000E011622DCAFF0000526362000000000000A0008029896F50
0000087000000000
```

L'exemple de balise HLS suivant montre comment utiliser EXT-X-CUE-OUT :

```
#EXT-OATCLS-SCTE35:/DA0AAAAAAAAAAAAABQb+ADAQ6QAeAhxDVUVJQAAA03/PAAEUrEoICAAAAAaG
+2UBNAAANvrtoQ==
#EXT-X-ASSET:CAID=0x0000000020FB6501
#EXT-X-CUE-OUT:201.467
```

L'exemple de balise HLS suivant montre comment utiliser EXT-X-SPLICEPOINT-SCTE35 :

```
#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAXMTMyMjE5M190NAAAmXz5JA==
```

L'exemple suivant montre comment utiliser le scte35:Binary décodage :

```
{
  "table_id": 252,
  "section_syntax_indicator": false,
  "private_indicator": false,
  "section_length": 33,
  "protocol_version": 0,
  "encrypted_packet": false,
  "encryption_algorithm": 0,
  "pts_adjustment": 0,
  "cw_index": 0,
  "tier": "0xFFF",
  "splice_command_length": 16,
  "splice_command_type": 5,
  "splice_command": {
    "splice_event_id": 448,
    "splice_event_cancel_indicator": false,
    "out_of_network_indicator": true,
    "program_splice_flag": true,
    "duration_flag": true,
    "splice_immediate_flag": false,
    "utc_splice_time": {
```

```

    "time_specified_flag": false,
    "pts_time": null
  },
  "component_count": 0,
  "components": null,
  "break_duration": {
    "auto_return": false,
    "duration": {
      "pts_time": 2160000,
      "wall_clock_seconds": 24.0,
      "wall_clock_time": "00:00:24:000000"
    }
  },
  "unique_program_id": 49152,
  "avail_num": 0,
  "avails_expected": 0,
  "segment_num": 0,
  "segments_expected": 0,
  "sub_segment_num": 0,
  "sub_segments_expected": 0
},
"splice_descriptor_loop_length": 0,
"splice_descriptors": null,
"Scte35Exception": {
  "parse_status": "SCTE-35 cue parsing completed with 0 errors.",
  "error_messages": [],
  "table_id": 252,
  "splice_command_type": 5
}
}

```

MediaTailor variables de lecteur pour les requêtes ADS

AWS Elemental MediaTailor envoie les données reçues du lecteur à l'ADS lorsque vous configurez AWS Elemental MediaTailor pour spécifier `player_params.<query_parameter_name>` des variables dans l'URL ADS modèle. Par exemple, si le joueur envoie un paramètre de requête nommé `user_id` dans sa demande à MediaTailor, pour transmettre ces données dans la demande ADS, incluez-le `[player_params.user_id]` dans la configuration de l'URL ADS.

Cela vous permet de contrôler les paramètres de la requête qui sont inclus dans la demande ADS. En général, vous ajoutez un paramètre de requête particulier que le serveur ADS reconnaît à l'URL de demande du serveur ADS et fournissez les paires clé-valeur comme la valeur du paramètre.

Les exemples utilisés dans la procédure suivante utilisent les paires clé-valeur suivantes :

- param1 avec une valeur valeur1 :
- param2 avec une valeur valeur2 :

Pour ajouter les paramètres de requête en tant que paires clé-valeur

1. Dans AWS Elemental MediaTailor, configurez l'URL du modèle de demande ADS pour référencer les paramètres. L'URL suivante affiche l'inclusion des exemples de paramètres :

```
https://my.ads.com/path?param1=[player_params.param1]&param2=[player_params.param2]
```

2. (Facultatif) Pour les rapports de suivi des publicités côté serveur, URL-encode les paires clé-valeur sur le lecteur. Lorsqu'il MediaTailor reçoit la demande d'initialisation de session, il saisit URL-decodes les valeurs une fois avant de les substituer dans l'URL de la demande ADS.

 Note

Si votre ADS nécessite une URL-encoded valeur, celle-ci URL-encode doit être doublée sur le lecteur. De cette façon, le décodage effectué par MediaTailor aboutit à une valeur codée une fois pour l'ADS.

Par exemple, si la représentation décodée des valeurs envoyées à l'ADS est `param1=value1:&param2=value2:`, alors la URL-encoded représentation l'est `param1=value1%3A&param2=value2%3A`.

3. Lors de l'appel d'initialisation de session depuis le lecteur, transmettez les paires clé-valeur à MediaTailor la valeur d'un seul paramètre de requête. Les exemples d'appels suivants fournissent les exemples de paires clé-valeur pour le reporting du suivi publicitaire côté serveur et côté client.
- Exemples de demandes de rapports de suivi des publicités côté serveur, utilisant des paires URL-encoded

HLS :

```
<master>.m3u8?ads.param1=value1%3A&ads.param2=value2%3A
```

DASH :

```
<manifest>.mpd?ads.param1=value1%3A&ads.param2=value2%3A
```

- Exemple de demande de rapports de suivi des publicités côté client, sans URL-encoding

HLS :

```
POST <master>.m3u8
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

DASH :

```
POST <manifest>.mpd
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

Pour les rapports côté serveur, MediaTailor décode les paramètres lorsque la demande du joueur est reçue. Pour les rapports côté client, cela ne modifie pas les paramètres reçus dans la charge utile JSON. MediaTailor envoie la demande suivante à l'ADS :

```
https://my.ads.com/<path>?param1=value1:&param2=value2:
```

De cette manière, les paires clé-valeur param1 et param2 sont incluses en tant que paramètres de requête de première classe dans la demande ADS.

MediaTailor variables de domaine pour plusieurs sources de contenu

AWS Elemental MediaTailor les variables de domaine dynamiques vous permettent d'utiliser plusieurs domaines, tels que la partie my-ads-server.com de l'URL `http://my-ads-server.com`, avec les paramètres du lecteur dans votre configuration. Cela vous permet d'utiliser plusieurs sources de contenu ou serveur de décision publicitaire (ADS) dans une seule configuration.

Vous pouvez utiliser des variables de domaine avec n'importe quel paramètre contenant un URI :

- `AdDecisionServerUrl`
- `AdSegmentUrlPrefix`
- `ContentSegmentUrlPrefix`
- `LivePreroll.AdDecisionServerUrl`
- `VideoContentSourceUrl`

Les variables de domaine sont utilisées conjointement avec les alias de configuration pour effectuer le remplacement dynamique des variables. Les alias de configuration font correspondre un ensemble d'alias et de valeurs aux paramètres du lecteur utilisés pour la configuration dynamique du domaine. Pour les procédures de configuration, consultez [Création et utilisation d'alias de configuration avec MediaTailor](#). Pour des informations de référence détaillées, consultez [MediaTailor vue d'ensemble des alias de configuration](#).

MediaTailor variables de service pour le contrôle de session

AWS Elemental MediaTailor réserve l'espace de noms des paramètres de `aws.` requête pour les variables de service qui contrôlent le comportement au niveau de la session. Contrairement aux `ads.` paramètres (qui sont transmis à l'ADS) et aux `manifest.` paramètres (qui sont ajoutés aux URL du manifeste personnalisé), les `aws.` paramètres sont consommés directement par le serveur d'origine ou l'ADS MediaTailor et ne sont pas transmis à celui-ci.

Paramètres pris en charge

Le tableau suivant répertorie les `aws.*` paramètres que vous pouvez utiliser pour contrôler le comportement au niveau de la session.

Paramètre	Type	Valeurs	Par défaut	Description
<code>aws.start Time</code>	Horodatage ISO 8601	Par exemple, <code>2026-06-17T10:00:00Z</code>	Non défini (jointure live-edge)	Démarre la session à un endroit précis de la fenêtre du DVR. MediaTailor résout l'horodatage à la limite de segment la plus proche et émet <code>EXT-X-START:TIME-OFFSET</code> dans le manifeste HLS.

Paramètre	Type	Valeurs	Par défaut	Description
<code>aws.preroll</code>	String enum	<code>enabled</code> , <code>disabled</code> (sans distinction majuscules/ minuscules)	<code>enabled</code>	Détermine si l'insertion d'annonces avant diffusion a lieu pour la session. Lorsque <code>disabled</code> , le pré-roll est supprimé même si la configuration de lecture comporte un <code>LivePreRollConfiguration</code> .
<code>aws.overlayAvails</code>	String enum	<code>on</code> , <code>off</code> (sans distinction majuscules/ minuscules)	<code>on</code>	Contrôle si la superposition (non linéaire) et les disponibilités sont traitées pour la session. Lorsque <code>off</code> les marqueurs publicitaires superposés dans le manifeste source sont ignorés et qu'aucune annonce superposée n'est insérée.
<code>aws.logMode</code>	String enum	<code>DEBUG</code> , <code>DISABLED</code>	<code>DISABLED</code>	Active une journalisation détaillée du débogage pour la session. Lorsque ce paramètre est défini sur <code>DEBUG</code> , MediaTailor envoie des journaux de session détaillés vers CloudWatch Logs à des fins de dépannage.
<code>aws.availSuppressionMode</code>	String enum	<code>OFF</code> , <code>BEHIND_LIVE_EDGE</code> , <code>AFTER_LIVE_EDGE</code> (sans distinction majuscules/ minuscules)	<code>OFF</code>	Contrôle si les publicités sont supprimées en fonction de leur position par rapport au Live Edge.

Paramètre	Type	Valeurs	Par défaut	Description
<code>aws.availSuppressionValue</code>	Durée	HH:MM:SS format (par exemple, 00:00:10)	Non défini	Le créneau horaire pour la suppression de la disponibilité. Obligatoire quand <code>availSuppressionMode</code> c'est <code>BEHIND_LIVE_EDGE</code> ou <code>AFTER_LIVE_EDGE</code> .
<code>aws.availSuppressionFillPolicy</code>	String enum	<code>FULL_AVAI</code> <code>L_ONLY</code> , <code>PARTIAL_AVAI</code> (sans distinction majuscules/minuscules)	<code>FULL_AVAI</code> <code>L_ONLY</code>	Lorsque le mode est <code>AFTER_LIVE_EDGE</code> défini, contrôle si les valeurs partiellement supprimées sont remplies.

Heure de début AWS

Quand `aws.startTime` est défini, MediaTailor démarre la session à la limite du segment la plus proche de la date et de l'heure du programme spécifié.

Usage

Passez `aws.startTime` en tant que paramètre de requête dans la requête manifeste :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?
aws.startTime=2026-06-17T10:00:00Z
```

Ou lors d'une initialisation explicite de session, transmettez-le en tant que champ de niveau supérieur sans le `aws.` préfixe :

```
POST /v1/session/{hashed-account-id}/{origin-id}/{asset}.m3u8

{
  "startTime": "2026-06-17T10:00:00Z"
}
```

Exigences

Les exigences suivantes s'appliquent à `aws.startTime` :

- Le manifeste source doit contenir EXT-X-PROGRAM-DATE-TIME (PDT) sur les segments. Sans PDT, il `aws.startTime` ne peut pas être résolu et est ignoré.
- S'applique uniquement aux sessions HLS live (SSAI et SGAI).

Comportement

Le tableau suivant décrit le `aws.startTime` comportement dans différents scénarios :

Scénario	Résultat
Horodatage dans la fenêtre du DVR	Commencez à la limite du segment le plus proche, émettez EXT-X-START
Horodatage antérieur à la fenêtre du DVR	Fixez à 3 fois la durée de la cible à partir de la tête de la fenêtre
Horodatage inférieur à 3 fois la durée cible et la durée du live edge	Traiter comme une jointure en direct (non) EXT-X-START
Horodatage à ou après Live Edge	Jointure Live-Edge normale
Malformé ou non ISO 8601	Ignoré : retour à la jointure live-edge, erreur enregistrée
Le manifeste n'a pas de PDT	Ignoré : retour à la jointure live-edge, erreur enregistrée
Paramètre omis	Par défaut : jointure live-edge normale

Comportement de serrage

Si le point de départ dépasse la fenêtre du DVR en milieu de session, il EXT-X-START est limité à 3 fois la durée de la cible par rapport à la tête de la fenêtre. La mémoire tampon $3 \times \text{TargetDuration}$ est conforme aux recommandations de mise en mémoire tampon des joueurs de la RFC 8216 §6.3.3.

Exemple Sortie du manifeste avec AWS.StartTime

Lorsqu'un joueur initialise une session avec `aws.startTime=2026-06-17T10:00:00Z` un décalage résolu à 120 secondes du Live Edge, il MediaTailor émet :

```
#EXTM3U
#EXT-X-TARGETDURATION:6
#EXT-X-START:TIME-OFFSET=-120.120,PRECISE=YES
#EXT-X-MEDIA-SEQUENCE:500
#EXT-X-PROGRAM-DATE-TIME:2026-06-17T09:58:00.000Z
#EXTINF:6.006,
segment500.ts
#EXTINF:6.006,
segment501.ts
...
```

`TIME-OFFSET=-120.120` Indique au joueur de commencer la lecture 120 secondes après le Live Edge, à la limite du segment la plus proche de l'heure de début demandée.

Limitations

Les limites suivantes s'appliquent :

- Vous ne pouvez pas modifier ce paramètre après l'initialisation de la session.
- Nécessite `EXT-X-PROGRAM-DATE-TIME` dans le manifeste source.
- `EXT-X-START` est un indice pour les joueurs. Nous MediaTailor ne pouvons pas garantir que tous les joueurs l'honoreront.

aws.preroll

Lorsque `aws.preroll=disabled`, MediaTailor supprime l'insertion d'annonces avant diffusion pour la session, même si la configuration de lecture comporte un `LivePreRollConfiguration`.

Usage

Passez `aws.preroll` en tant que paramètre de requête dans la requête manifeste :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?aws.preroll=disabled
```

Ou lors d'une initialisation explicite de session, transmettez-le en tant que champ de niveau supérieur sans le `aws.` préfixe :

```
POST /v1/session/{hashed-account-id}/{origin-id}/{asset}.m3u8
{
```

```
"preroll": "disabled"
}
```

Exigences

Les exigences suivantes s'appliquent à `aws.preroll` :

- La configuration de lecture doit comporter un `LivePreRollConfiguration` pour que ce paramètre ait un effet. Si aucun pré-roll n'est configuré, la définition de ce paramètre n'a aucun effet.
- S'applique aux sessions HLS live (SSAI et SGAI).

Comportement

Le tableau suivant décrit comment `aws.preroll` se comporte :

Value	Résultat
enabled(ou omis)	Pre-roll inséré comme d'habitude
disabled	Pre-roll supprimé pour cette session
Valeur non valide	Consigné comme une erreur, traité comme enabled

Limitations

Vous ne pouvez pas modifier ce paramètre après l'initialisation de la session.

AWS. Overlay est disponible

Lorsque `aws.overlayAvails=off`, MediaTailor ignore les marqueurs publicitaires superposés (non linéaires) dans le manifeste source et n'insère pas de publicités superposées pour la session.

Usage

Passez `aws.overlayAvails` en tant que paramètre de requête dans la requête manifeste :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?aws.overlayAvails=off
```

Exigences

Les exigences suivantes s'appliquent à `aws.overlayAvails` :

- Le manifeste source doit contenir des marqueurs publicitaires superposés (par exemple, SCTE-35 des événements dont le type de segmentation est superposé) pour que ce paramètre ait un effet.
- S'applique aux sessions HLS et DASH, en direct et à la VOD (SSAI et SGAI).

Comportement

Le tableau suivant décrit comment `aws.overlayAvails` se comporte :

Value	Résultat
on(ou omis)	Les superpositions disponibles sont traitées et les publicités sont insérées comme d'habitude
off	Les marqueurs publicitaires superposés sont ignorés, aucune annonce superposée n'est insérée
Valeur non valide	Consigné comme erreur, traité comme non spécifié (par défaut :on)

Limitations

Vous ne pouvez pas modifier ce paramètre après l'initialisation de la session.

AWS. Log Mode

Quand `aws.logMode=DEBUG`, MediaTailor active la journalisation détaillée du débogage pour la session. Les journaux de débogage sont envoyés aux CloudWatch journaux et fournissent des informations détaillées sur la personnalisation des manifestes, les demandes du serveur de décision en matière d'annonces et l'état de la session, ce qui est utile pour résoudre les problèmes d'insertion d'annonces.

Usage

Passez `aws.logMode` en tant que paramètre de requête dans la requête manifeste :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?aws.logMode=DEBUG
```

Exigences

Les exigences suivantes s'appliquent à `aws.LogMode` :

- La journalisation doit être activée (`PercentEnabled > 0` ou `EnabledLoggingStrategies` configurée) dans la configuration de lecture pour que les journaux de débogage soient émis.
- La journalisation des débogages est limitée par client afin d'éviter un volume de journaux excessif.

Comportement

Le tableau suivant décrit comment `aws.LogMode` se comporte :

Value	Résultat
DEBUG	Journaux de débogage détaillés émis pour la session
DISABLED(ou omis)	Comportement de journalisation normal (basé sur les paramètres de configuration de lecture)
Valeur non valide	Erreur générée, échec de l'initialisation de la session

Limitations

Les limites suivantes s'appliquent :

- Vous ne pouvez pas modifier ce paramètre après l'initialisation de la session.
- Les valeurs distinguent les majuscules et minuscules (DEBUGnondebug).

`aws.avail` `SuppressionMode`

Contrôle si les publicités sont supprimées en fonction de leur position par rapport au Live Edge. Utilisez cette option pour ignorer les pauses publicitaires qui se situent dans la fenêtre de temps précédant ou suivant le Live Edge, par exemple, pour éviter de remplir les pauses publicitaires que les spectateurs ont déjà passées lorsqu'ils ont rejoint une diffusion en direct en plein milieu d'un événement.

Usage

Transmettez les paramètres de suppression de disponibilité en tant que paramètres de requête dans la demande manifeste :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?  
aws.availSuppressionMode=BEHIND_LIVE_EDGE&aws.availSuppressionValue=00:00:10
```

Ce paramètre fonctionne avec deux paramètres complémentaires :

- `aws.availSuppressionValue`— La fenêtre temporelle (obligatoire lorsque le mode n'est pas activé `OFF`)
- `aws.availSuppressionFillPolicy`— Contrôle le comportement de remplissage partiel (s'applique uniquement au `AFTER_LIVE_EDGE` mode)

Exigences

Les exigences suivantes s'appliquent pour bénéficier des paramètres de suppression :

- S'applique aux sessions en direct HLS et DASH.
- `aws.availSuppressionValue` doit être fourni au HH:MM:SS format lorsque le mode est `BEHIND_LIVE_EDGE` ou `AFTER_LIVE_EDGE`.
- `aws.availSuppressionFillPolicy` n'est valide que lorsque le mode est activé `AFTER_LIVE_EDGE`.

Comportement du mode

Le tableau suivant décrit l'effet de chaque mode de suppression :

Mode	Effet
OFF(ou omis)	Suppression inutile : toutes les pauses publicitaires sont remplies normalement
BEHIND_LIVE_EDGE	Supprimez les pauses publicitaires qui commencent dans la fenêtre de temps spécifiée, après le Live Edge
AFTER_LIVE_EDGE	Supprimez les pauses publicitaires qui commencent après la fenêtre de temps spécifiée depuis Live Edge

Politique de remplissage (AFTER_LIVE_EDGE uniquement)

Le tableau suivant décrit le comportement de la politique de remplissage lorsque le mode est AFTER_LIVE_EDGE :

Politique de remplissage	Effet
FULL_AVAIL_ONLY (par défaut)	Ne remplissez que les zones qui se trouvent complètement en dehors de la fenêtre de suppression
PARTIAL_AVAIL	Remplissez la partie d'un champ qui s'étend au-delà de la fenêtre de suppression

Exemple Exemple de suppression de Avail

La requête suivante supprime les interruptions de publicité dans les 10 secondes qui suivent le Live Edge :

```
GET /v1/master/{hashed-account-id}/{origin-id}/{asset}.m3u8?  
aws.availSuppressionMode=BEHIND_LIVE_EDGE&aws.availSuppressionValue=00:00:10
```

Limitations

Les limites suivantes s'appliquent :

- Vous ne pouvez pas modifier ce paramètre après l'initialisation de la session.
- `aws.availSuppressionValue` ne doit pas être fourni lorsque le mode est activé OFF.
- Format d'heure non valide `aws.availSuppressionValue` dans lequel le mode est rétabli OFF.

MediaTailor vue d'ensemble des alias de configuration

AWS Elemental MediaTailor les alias de configuration permettent le remplacement dynamique des variables dans les domaines URL et autres champs pris en charge. Utilisez cette fonctionnalité pour utiliser plusieurs domaines et effectuer une configuration dynamique URLs lors de l'initialisation de la session.

Cas d'utilisation

Les alias de configuration permettent d'utiliser des architectures multiconfigurations sophistiquées pour les scénarios suivants :

- Routage géographique : acheminez les demandes vers différentes origines ou vers différents serveurs publicitaires en fonction de la localisation des internautes à l'aide d'alias spécifiques à la région. Pour obtenir des conseils de mise en œuvre, consultez [CloudFront Origin Failover](#).
- Routage basé sur le contenu : Dirigez les différents types de contenu vers des origines ou des pipelines de traitement spécialisés. Pour la configuration du comportement de routage, voir [the section called “Configurer les comportements de routage du CDN”](#).
- Scénarios de basculement : implémentez les origines de sauvegarde et les serveurs publicitaires avec un basculement automatique à l'aide du changement d'alias. Pour une mise en œuvre détaillée, voir [Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR](#) et [the section called “Planifier l'intégration du CDN”](#).
- Tests A/B : testez différents serveurs publicitaires, origines ou configurations en acheminant le trafic en fonction des paramètres du joueur. Pour obtenir des conseils sur les tests de charge, consultez [Préparer et exécuter des tests de performance pour Amazon CloudFront avec une véritable surveillance des utilisateurs](#).
- Optimisation spécifique à l'appareil : optimisez la diffusion de contenu et la diffusion d'annonces pour différents types d'appareils ou fonctionnalités. Pour des conseils complets, voir [Configurer le filtrage des manifestes avec MediaTailor MediaPackage, et le CDN](#).
- Équilibrage de charge : répartissez la charge entre plusieurs origines ou serveurs publicitaires à l'aide du routage dynamique. Pour des conseils de mise en œuvre, voir [Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR](#) et [the section called “Planifier l'intégration du CDN”](#).

Champs pris en charge

Vous pouvez utiliser des variables dynamiques dans les champs de configuration suivants :

- VideoContentSourceUrl
- AdDecisionServerUrl
- LivePreroll.AdDecisionServerUrl
- AdSegmentUrlPrefix
- ContentSegmentUrlPrefix

- `TranscodeProfileName`
- `SlateAdUrl`
- `StartUrl`
- `EndUrl`

Les sections suivantes décrivent comment utiliser les alias de configuration.

Rubriques

- [Création et utilisation d'alias de configuration avec MediaTailor](#)
- [Alias de configuration avec exemple MediaTailor d'utilisation](#)

Création et utilisation d'alias de configuration avec MediaTailor

Avant de commencer à utiliser des variables de domaine, vous devez créer des alias de configuration pour votre configuration. Vous utilisez les alias de configuration comme variables de remplacement de domaine au moment de l'initialisation de la session.

Restrictions

Notez les restrictions suivantes lorsque vous utilisez des alias de configuration :

- Toutes les variables dynamiques utilisées dans le domaine doivent être définies en tant que variables `ConfigurationAliases` dynamiques.
- Les variables des paramètres du joueur doivent être préfixées par `player_params..` Par exemple, `player_params.origin_domain`.
- La liste des valeurs aliasées doit être exhaustive pour les variables de domaine dans critical URLs (`VideoContentSourceUrl`, `AdSegmentUrlPrefix`, `ContentSegmentUrlPrefix`).
- Si une demande est faite pour une variable de domaine critique URLs qui ne spécifie pas la variable dynamique ou utilise un alias non valide, la demande échouera avec un code d'état HTTP 400. Les champs non critiques (`SlateAdUrl`, `TranscodeProfileName`, bumper URLs) enregistreront les avertissements mais ne feront pas échouer la demande.

Comportement de secours en cas d'alias manquants

Lorsque les alias de configuration sont introuvables ou ne sont pas valides, MediaTailor implémente le comportement de secours suivant :

- Variables de domaine : si un alias de variable de domaine est manquant ou non valide, la demande échoue avec le code d'état HTTP 400. Des alias valides doivent être définis pour toutes les variables de domaine.
- Variables non liées au domaine : pour les variables utilisées dans des parties non liées au domaine de URLs (telles que les éléments de chemin ou les paramètres de requête), les alias manquants entraînent le remplacement de chaînes vides.
- Validation de la configuration : MediaTailor vérifie que tous les alias requis sont présents lors des opérations de création et de mise à jour de la configuration.

Étape 1 : créer des alias de configuration

Pour créer des alias de configuration à utiliser pour le remplacement de domaines à l'aide de la MediaTailor console, effectuez la procédure suivante.

Console

Pour créer des alias de configuration à l'aide de la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans la section Alias de configuration de la page Configurations, choisissez Ajouter un paramètre de joueur.
3. Dans Paramètre du joueur, entrez le nom du paramètre du joueur que vous souhaitez utiliser comme variable dynamique. Par exemple, `player_params.origin_domain`.
4. Dans Alias, entrez les alias et leurs valeurs que vous souhaitez utiliser pour le paramètre du joueur.
5. Choisissez OK.

AWS Elemental MediaTailor affiche le nouveau paramètre dans le tableau de la section Alias de configuration.

6. Répétez les étapes précédentes pour ajouter d'autres paramètres au joueur.
7. Choisissez Enregistrer.

API

Pour créer des alias de configuration à l'aide de l'API

Lorsque vous créez ou mettez à jour une MediaTailor configuration, utilisez le `ConfigurationAliases` paramètre avec la structure JSON suivante :

```
{
    "ConfigurationAliases": {
        "player_params.origin_domain": {
            "pdx": "abc.mediapackage.us-west-2.amazonaws.com",
            "iad": "xyz.mediapackage.us-east-1.amazonaws.com"
        },
        "player_params.ad_type": {
            "customized": "abc12345",
            "default": "defaultAdType"
        }
    }
}
```

Étape 2 : Utiliser des alias de configuration lors de l'initialisation de session

Après avoir configuré les alias de configuration, vous pouvez les utiliser comme variables de remplacement pour les domaines dans votre demande d'initialisation de session. Cela vous permet de configurer dynamiquement les domaines de votre session.

Exemple Exemple d'alias de configuration de base

Voici un exemple de configuration de base qui inclut des alias de configuration et des variables de domaine dynamiques :

```
PUT /playbackConfiguration
{
    "Name": "aliasedConfig",
    "AdDecisionServerUrl": "https://abc.execute-api.us-west-2.amazonaws.com/ads?
sid=[session.id]&ad_type=[player_params.ad_type]",
    "VideoContentSourceUrl": "https://[player_params.origin_domain].mediapackage.
[player_params.region].amazonaws.com/out/v1/[player_params.endpoint_id]",
    "ConfigurationAliases": {
        "player_params.origin_domain": {
            "pdx": "abc",
            "iad": "xyz"
        },
        "player_params.region": {
            "pdx": "us-west-2",
            "iad": "us-east-1"
        }
    }
}
```

```
    },
    "player_params.endpoint_id": {
      "pdx": "abcd",
      "iad": "wxyz"
    },
    "player_params.ad_type": {
      "customized": "abc12345",
      "default": "defaultAdType"
    }
  }
}
```

Exemple Initialisation de session avec des alias

En utilisant la configuration précédente, une demande d'initialisation de session utilisant les variables et les alias du lecteur ressemblerait à ce qui suit :

```
POST index.m3u8
{
  "playerParams": {
    "origin_domain": "pdx",
    "region": "pdx",
    "endpoint_id": "pdx",
    "ad_type": "customized"
  }
}
```

MediaTailor remplace les chaînes d'alias par les valeurs mappées dans la configuration des alias de configuration.

La demande adressée à l'ADS ressemblera à ce qui suit :

```
https://abc.execute-api.us-west-2.amazonaws.com/ads?sid=[session.id]&ad_type=abc12345
```

La demande adressée à l'origine pour les manifestes ressemblera à ce qui suit :

```
https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd
```

Alias de configuration avec exemple MediaTailor d'utilisation

Les exemples suivants illustrent une MediaTailor configuration complète avec des alias de configuration, une demande d'initialisation de session avec des alias et le flux de traitement des alias.

Exemple Configuration complète avec alias

L'exemple suivant montre une configuration complète qui inclut des alias de configuration et des variables de domaine dynamiques :

```
PUT /playbackConfiguration
{
  "Name": "aliasedConfig",
  "AdDecisionServerUrl": "https://abc.execute-api.us-west-2.amazonaws.com/ads?
sid=[session.id]&ad_type=[player_params.ad_type]",
  "VideoContentSourceUrl": "https://[player_params.origin_domain].mediapackage.
[player_params.region].amazonaws.com/out/v1/[player_params.endpoint_id]",

  "AdSegmentUrlPrefix": "https://[player_params.ad_cdn_domain]/ads/",
  "ContentSegmentUrlPrefix": "https://[player_params.content_cdn_domain]/content/",
  "TranscodeProfileName": "[player_params.transcode_profile]",
  "SlateAdUrl": "https://[player_params.slate_domain]/slate/
[player_params.slate_type].mp4",
  "StartUrl": "https://[player_params.tracking_domain]/start?session=[session.id]",
  "EndUrl": "https://[player_params.tracking_domain]/end?session=[session.id]",

  "ConfigurationAliases": {
    "player_params.origin_domain": {
      "pdx": "abc",
      "iad": "xyz"
    },
    "player_params.region": {
      "pdx": "us-west-2",
      "iad": "us-east-1"
    },
    "player_params.endpoint_id": {
      "pdx": "abcd",
      "iad": "wxyz"
    },
    "player_params.ad_type": {
      "customized": "abc12345",
      "default": "defaultAdType"
    },
    "player_params.ad_cdn_domain": {
      "pdx": "ads-west.cdn.example.com",
      "iad": "ads-east.cdn.example.com"
    },
    "player_params.content_cdn_domain": {
      "pdx": "content-west.cdn.example.com",
```



```
        "iad": "content-east.cdn.example.com"
    },
    "player_params.transcode_profile": {
        "mobile": "mobile_optimized",
        "desktop": "high_quality",
        "tv": "4k_profile"
    },
    "player_params.slate_domain": {
        "pdx": "slate-west.example.com",
        "iad": "slate-east.example.com"
    },
    "player_params.slate_type": {
        "standard": "default_slate",
        "branded": "brand_slate"
    },
    "player_params.tracking_domain": {
        "pdx": "tracking-west.example.com",
        "iad": "tracking-east.example.com"
    }
}
}
```

Exemple Initialisation de session avec des alias

L'exemple suivant montre une demande d'initialisation de session qui spécifie les variables et alias du joueur :

```
POST master.m3u8
{
  "playerParams": {
    "origin_domain": "pdx",
    "region": "pdx",
    "endpoint_id": "pdx",
    "ad_type": "customized",
    "ad_cdn_domain": "pdx",
    "content_cdn_domain": "pdx",
    "transcode_profile": "mobile",
    "slate_domain": "pdx",
    "slate_type": "branded",
    "tracking_domain": "pdx"
  }
}
```

Exemple Flux de traitement des paramètres

Dans l'exemple suivant, MediaTailor remplace les chaînes d'alias par les valeurs mappées dans les alias de configuration. Le traitement donne lieu aux demandes suivantes :

- Demande ADS :

```
https://abc.execute-api.us-west-2.amazonaws.com/ads?sid=[session.id]&ad_type=abc12345
```

- VideoContentSource demande :

```
https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd
```

- AdSegmentUrlPrefix:

```
https://ads-west.cdn.example.com/ads/
```

- ContentSegmentUrlPrefix:

```
https://content-west.cdn.example.com/content/
```

- TranscodeProfileName:

```
mobile_optimized
```

- SlateAdUrl:

```
https://slate-west.example.com/slate/brand_slate.mp4
```

- StartUrl:

```
https://tracking-west.example.com/start?session=[session.id]
```

- EndUrl:

```
https://tracking-west.example.com/end?session=[session.id]
```

MediaTailor transmission de paramètres à ADS

AWS Elemental MediaTailor prend en charge la configuration de variables dynamiques dans les MediaTailor requêtes adressées à l'ADS en suivant les étapes suivantes.

- Pour plus d'informations sur le formatage pris en charge pour les paramètres de requête, consultez [MediaTailor référence des paramètres et limites](#).
- Pour les alias de configuration et les variables de domaine, consultez [MediaTailor vue d'ensemble des alias de configuration](#).
- Pour des personnalisations supplémentaires de la demande ADS, consultez [Utilisation avancée](#).

Méthodes d'initialisation de session

MediaTailor prend en charge plusieurs méthodes d'initialisation de session et de transmission de paramètres :

1. POST avec corps de demande :

```
POST <master>.m3u8
{
  "adsParams": {"param1": "value1", "param2": "value2"},
  "playerParams": {"param3": "value3"}
}
```

2. Paramètres de requête dans l'URL :

```
GET <master>.m3u8?ads.param1=value1&ads.param2=value2&playerParams.param3=value3
```

Important

Vous ne pouvez spécifier les paramètres qu'une seule fois, au moment de l'initialisation. Les alias de configuration sont résolus en valeurs réelles avant le transfert.

Pour transmettre les informations de session et de lecteur au serveur ADS

1. Travaillez avec l'ADS pour déterminer les informations dont il a besoin pour répondre à une requête publicitaire émanant de AWS Elemental MediaTailor.

2. Créez une configuration MediaTailor qui utilise un modèle d'URL de demande ADS qui répond aux exigences ADS. Dans l'URL, incluez les paramètres statiques et les espaces réservés correspondant aux paramètres dynamiques. Entrez l'URL de votre modèle dans le champ Ad decision server (Serveur ADS) de la configuration.

Dans l'exemple suivant d'URL du modèle `correlation` fournit les données de session et `deviceType` fournit des données de lecteur :

```
https://my.ads.server.com/path?  
correlation=[session.id]&deviceType=[player_params.deviceType]
```

3. Sur le joueur, configurez la demande d'initiation de session pour que AWS Elemental MediaTailor fournisse les paramètres pour les données de lecteur. Incluez vos paramètres dans la demande d'initiation de session et omettez-les des demandes suivantes pour la session.

Le type d'appel que le joueur effectue pour initialiser la session détermine si le joueur (client) ou MediaTailor (serveur) fournit des rapports de suivi des publicités pour la session. Pour plus d'informations sur ces deux options, consultez la section [Rapports et données de suivi](#).

Effectuez l'une des types suivants d'appels, selon que vous voulez un rapport de suivi publicitaire côté serveur ou côté client. Dans les deux exemples d'appels, `userID` est destiné au serveur ADS et `auth_token` à l'origine :

- (Option) Demandez des rapports de suivi des publicités côté serveur — Préfixez les paramètres avec lesquels vous souhaitez envoyer MediaTailor à l'ADS. `ads` Conservez le préfixe désactivé pour les paramètres que MediaTailor doit envoyer au serveur d'origine :

Les exemples suivants montrent les demandes entrantes adressées à HLS et DASH. AWS Elemental MediaTailor utilise le `deviceType` dans sa demande à l'ADS et le `auth_token` dans sa demande au serveur d'origine.

Exemple HLS :

```
GET master.m3u8?ads.deviceType=ipad&auth_token=kjhdsaf7gh
```

Exemple DASH :

```
GET manifest.mpd?ads.deviceType=ipad&auth_token=kjhdsaf7gh
```

- (Option) Demandez des rapports de suivi des publicités côté client : fournissez les paramètres de l'ADS à l'intérieur d'un objet. `adsParams`

Exemple HLS :

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

Exemple DASH :

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

Lorsque le joueur lance une session, AWS Elemental MediaTailor remplace les variables du modèle d'URL de demande ADS par les données de session et les ads paramètres du joueur. Il transmet les paramètres restants du lecteur au serveur d'origine.

Exemple MediaTailor requêtes avec variables publicitaires

Les exemples suivants illustrent les appels à l'ADS et au serveur d'origine provenant d' AWS Elemental MediaTailor qui correspondent aux exemples d'appel d'initialisation de session de lecteur précédents :

- MediaTailor appelle l'ADS avec les données de session et le type d'appareil du joueur :

```
https://my.ads.server.com/path?correlation=896976764&deviceType=ipad
```

- MediaTailor appelle le serveur d'origine avec le jeton d'autorisation du joueur.

- Exemple HLS :

```
https://my.origin.server.com/master.m3u8?auth_token=kjhdsaf7gh
```

- Exemple DASH :

```
https://my.origin.server.com/manifest.mpd?auth_token=kjhdsaf7gh
```

Utilisation avancée

Vous pouvez personnaliser la demande ADS de multiples façons avec les données du lecteur et les données de session. Il vous suffit d'inclure le nom d'hôte ADS.

Les exemples suivants illustrent certaines façons de personnaliser votre demande :

- Concaténez les paramètres du lecteur et les paramètres de session pour créer de nouveaux paramètres. Exemple :

```
https://my.ads.com?key1=[player_params.value1][session.id]
```

- Utilisez un paramètre du lecteur comme partie intégrante d'un élément du chemin. Exemple :

```
https://my.ads.com/[player_params.path]?key=value
```

- Utilisez les paramètres du lecteur pour transmettre les éléments du chemin d'accès et les clés elles-mêmes, plutôt que les seules valeurs. Exemple :

```
https://my.ads.com/[player_params.path]?[player_params.key1]=[player_params.value1]
```

MediaTailor routage des paramètres pour ADS et origines

AWS Elemental MediaTailor achemine les paramètres de requête vers différentes destinations en fonction de leur préfixe et de leur objectif. Il est essentiel de comprendre le routage des paramètres pour implémenter des fonctionnalités spécifiques à l'origine, telles que la visualisation décalée dans le temps avec MediaPackage.

Règles de routage des paramètres

MediaTailor utilise les règles de routage suivantes pour les paramètres de requête :

- Paramètres d'origine (pas de préfixe) : les paramètres sans préfixe spécifique sont transmis au serveur d'origine pour des fonctionnalités spécifiques à l'origine.

- Paramètres ADS (**ads.**préfixe) : les paramètres préfixés par `ads.` sont envoyés au serveur Ad Decision Server
- Paramètres du manifeste (**manifest.**préfixe) : les paramètres préfixés par `manifest.` sont utilisés pour le routage et l'autorisation du CDN

Exemple Exemple de routage des paramètres

L'initialisation de session suivante illustre le routage des paramètres :

```
POST /v1/session/123456789/originId/index.m3u8
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2"
  },
  "manifestParams": {
    "auth_token": "abc123"
  }
}
```

Dans cet exemple :

- `param1` et `param2` sont envoyés à l'ADS
- `auth_token` est utilisé pour le routage et l'autorisation du CDN
- Les paramètres sans préfixes seraient transmis au serveur d'origine

Comportement des paramètres du serveur Origin

Les paramètres transmis aux serveurs d'origine activent des fonctionnalités spécifiques à l'origine, telles que l'affichage décalé dans le temps, le filtrage du contenu et l'authentification.

Cas d'utilisation courants des paramètres d'origine

Les paramètres d'origine sont couramment utilisés pour :

- Time-shifted affichage : `start` et `end` paramètres pour le contenu MediaPackage décalé dans le temps
- Authentification du contenu : jetons d'authentification requis par le serveur d'origine
- Filtrage du contenu : paramètres qui contrôlent les variantes de contenu renvoyées

- Origin-specific fonctionnalités : tous les paramètres utilisés par le serveur d'origine pour le traitement du contenu

 Important

Les paramètres sont traités lors de l'initialisation de la session et conservés tout au long de la session. Pour modifier des paramètres tels que les fenêtres de décalage temporel, vous devez créer une nouvelle session avec des valeurs mises à jour.

MediaTailor et intégration de visualisation MediaPackage décalée dans le temps

AWS Elemental MediaTailor peut transmettre des paramètres de visualisation décalés dans le temps aux MediaPackage origines pour permettre la fonctionnalité de visualisation de redémarrage et de rattrapage. Cette intégration permet aux spectateurs de commencer à regarder du contenu en direct plus tôt dans le temps.

MediaPackage paramètres de visualisation décalés dans le temps

MediaPackage prend en charge les paramètres de visualisation décalés dans le temps suivants qui peuvent être transmis MediaTailor :

- `start`: Epoch ou horodatage ISO 8601 définissant le début du manifeste décalé dans le temps
- `end`: Epoch ou horodatage ISO 8601 définissant la fin du manifeste décalé dans le temps
- `time_delay`: retardez la disponibilité du contenu de certaines secondes
- `manifest_window_seconds`: demandez un manifeste plus court que la fenêtre configurée

Exemple MediaTailor initialisation de session avec des paramètres décalés MediaPackage dans le temps

L'exemple suivant montre comment initialiser une session avec des paramètres d'affichage décalés dans le temps :

```
GET /v1/master/123456789/originId/index.m3u8?  
start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```


Ou en utilisant une initialisation explicite de session :

```
POST /v1/session/123456789/originId/index.m3u8
{
  "adsParams": {
    "param1": "value1"
  }
}
```

Avec des paramètres de requête supplémentaires :

```
?start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```

Comportement des paramètres pendant les sessions

Time-shifted les paramètres de visualisation ont des caractéristiques de comportement spécifiques :

- Initialisation de session : les paramètres sont traités lors de la création de la session
- Persistance des paramètres : les paramètres restent associés à la session tout au long de la lecture
- Immuable après initialisation : les paramètres ne peuvent pas être modifiés pendant une session active
- Nouvelle session requise : pour modifier les fenêtres de décalage temporel, créez une nouvelle session avec des valeurs de paramètres mises à jour

MediaPackage exigences relatives à la fenêtre de démarrage

Pour que l'affichage décalé dans le temps fonctionne MediaPackage, assurez-vous de ce qui suit :

1. Configurez une fenêtre de démarrage sur votre MediaPackage terminal (jusqu'à 24 heures)
2. Assurez-vous que votre CDN transmet les paramètres de requête nécessaires à MediaPackage
3. Utilisez des fenêtres de lecture cohérentes entre les sessions de lecteur pour une meilleure mise en cache du CDN
4. Vérifiez que les heures de début et de fin se situent dans la fenêtre de démarrage configurée

⚠ Important

Lorsque vous utilisez le visionnage décalé dans le temps, utilisez des fenêtres de lecture cohérentes entre les sessions de lecture plutôt que de générer des heures de début ou de fin uniques pour chaque spectateur. Cela permet d'améliorer la mise en cache sur le CDN et d'éviter tout ralentissement potentiel.

Pour obtenir des informations complètes sur la configuration et les paramètres d'affichage MediaPackage décalé dans le temps, reportez-vous à la section [Time-shifted Visualisation avec AWS Elemental MediaPackage](#) dans le Guide de AWS Elemental MediaPackage l'utilisateur.

MediaTailor comportement et persistance des sessions de paramètres

AWS Elemental MediaTailor traite les paramètres lors de l'initialisation de la session et les conserve tout au long du cycle de vie de la session. Il est essentiel de comprendre le comportement des sessions pour mettre en œuvre des scénarios de paramètres dynamiques.

Méthodes d'initialisation de session

MediaTailor prend en charge plusieurs méthodes d'initialisation de session avec des paramètres :

1. Initialisation implicite de session : paramètres inclus dans la demande de manifeste initiale

```
GET /v1/master/123456789/originId/index.m3u8?
manifest.auth_token=abc123&start=2024-08-26T10:00:00Z
```

2. Initialisation explicite de session (POST) : paramètres fournis dans le corps de la requête

```
POST /v1/session/123456789/originId/index.m3u8
{
  "adsParams": {"param1": "value1"},
  "manifestParams": {"auth_token": "abc123"}
}
```

3. Initialisation explicite de session (GET) : paramètres fournis en tant que paramètres de requête

```
GET /v1/session/123456789/originId/index.m3u8?
ads.param1=value1&manifestParams.auth_token=abc123
```

Persistence et immuabilité des paramètres

MediaTailor le comportement des paramètres suit les règles suivantes :

- One-time spécification : Les paramètres ne peuvent être spécifiés qu'une seule fois, lors de l'initialisation de la session
- Session-wide persistance : les paramètres sont conservés pendant toute la durée de la session
- Immuable après initialisation : les paramètres ne peuvent pas être modifiés après la création de la session
- Résolution des alias de configuration : les alias sont résolus en valeurs réelles avant d'être transférés vers les destinations

Scénarios de modification des paramètres

Pour modifier les paramètres pendant la lecture :

- Créer une nouvelle session : initialisez une nouvelle session avec des valeurs de paramètres mises à jour
- Transition du joueur : passez facilement le joueur à la nouvelle session
- Héritage des paramètres : reportez les paramètres inchangés pour maintenir la cohérence

Exemple Modifier les paramètres de décalage temporel

Pour passer d'une fenêtre de 1 heure à une fenêtre de 2 heures :

1. Session en cours : `start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z`
2. Créez une nouvelle session : `start=2024-08-26T10:00:00Z&end=2024-08-26T12:00:00Z`
3. Transférer le lecteur vers une nouvelle URL de session

Important

Les demandes de playlist multivariantes multiples pour une seule session ne mettent pas à jour les paramètres après la première demande. Les paramètres restent immuables pendant toute la durée de la session.

MediaTailor référence des paramètres et limites

Avant de configurer des variables publicitaires dynamiques, comprenez les exigences et les limites de mise en forme des paramètres qui s'appliquent à toutes les MediaTailor configurations.

AWS Elemental MediaTailor fournit des informations complètes sur les restrictions relatives aux caractères des paramètres, les limites de longueur et les formats pris en charge pour les paramètres de requête du manifeste et les paramètres ADS.

Restrictions relatives aux caractères des paramètres de requête Manif

Les paramètres de requête du manifeste prennent en charge des caractères spécifiques et ont des limites de longueur définies.

Personnages pris en charge (sans URL-encoding)

Vous pouvez utiliser les caractères suivants directement dans les paramètres de requête du manifeste :

- Caractères alphanumériques (A-Z, a-z, 0-9)
- Périodes (.)
- Tirets (-)
- Soulignés (_)
- barres obliques inverses (\)

Personnages pris en charge avec URL-encoding

Les caractères spéciaux suivants sont pris en charge lorsque URL-encoded :

- période (.) = %2E
- tiret (-) = %2D
- trait de soulignement (_) = %5F
- pourcentage (%) = 25 %
- tilde (~) = %7E
- barre oblique (/) = %2F
- astérisque (*) = %2A
- est égal à (=) = %3D

- question (?) = 3 : F

URL-encoding soutien

MediaTailor prend en charge le signe de pourcentage (%) lorsque vous l'utilisez dans URL-encoding (par exemple, hello%20world = hello world). Vous pouvez utiliser tous URL-encoded les caractères, à condition qu'ils soient valides URL-encodings conformément à la spécification HTTP.

Personnages non pris en charge

Vous ne pouvez pas utiliser les caractères suivants dans les paramètres de requête du manifeste sans URL-encoding : : ?&=,%, / (barre oblique).

Important

MediaTailor ne prend pas en charge les caractères doubles tels que%% ou ==. Vous ne pouvez pas utiliser des URL complètes comme valeurs de paramètres de requête du manifeste en raison de restrictions de caractères.

Limites de longueur

La longueur totale de tous les paramètres de requête du manifeste (clés et valeurs combinées) ne doit pas dépasser 2 000 caractères.

Limites de longueur des paramètres ADS

Les limites de longueur suivantes s'appliquent aux paramètres utilisés dans les requêtes adressées à l'ADS :

- Nom du paramètre ADS : 10 000 caractères maximum
- Valeur du paramètre ADS : 25 000 caractères maximum
- URL ADS : 25 000 caractères maximum

MediaTailor guide de dépannage des paramètres

AWS Elemental MediaTailor fournit des conseils pour résoudre les problèmes courants liés aux paramètres, notamment les restrictions de caractères MediaTailor, les problèmes de codage des URL et les erreurs d'alias de configuration.

Erreurs de restriction de caractères

Les valeurs de paramètres contenant des caractères non pris en charge peuvent provoquer des erreurs ou des comportements inattendus.

Symptômes courants

Les symptômes suivants peuvent indiquer des problèmes de restriction des caractères :

- Paramètres n'apparaissant pas dans les URL du manifeste
- Erreurs HTTP 400 lors de l'initialisation de la session
- Valeurs de paramètres tronquées ou corrompues
- Les requêtes ADS échouent en raison d'URL mal formées

Étapes de résolution

Pour résoudre les erreurs de restriction de caractères :

1. Vérifiez les valeurs des paramètres pour les caractères non pris en charge : : ?, &, ,, =, % /
2. Appliquez correctement l'URL-encoding pour les caractères spéciaux (voir [MediaTailor référence des paramètres et limites](#))
3. Évitez les caractères doubles tels que %%% ou ==
4. Envisagez d'autres formats de paramètres si les URL complètes ne peuvent pas être utilisées

Exemple de codage d'URL

Au lieu d'utiliser :

```
manifest.redirect_url=https://example.com/path?param=value
```

URL-encoded Format d'utilisation :

```
manifest.redirect_url=https%3A%2F%2Fexample.com%2Fpath%3Fparam%3Dvalue
```

Erreurs de limitation de longueur

Les paramètres qui dépassent les limites de longueur peuvent être tronqués ou provoquer des erreurs.

Limites de longueur

Les limites de longueur suivantes s'appliquent (voir [MediaTailor référence des paramètres et limites](#) pour plus de détails) :

- Paramètres de requête du manifeste (total) : 2000 caractères
- Noms des paramètres ADS : 10 000 caractères
- Valeurs des paramètres ADS : 25 000 caractères
- URL ADS : 25 000 caractères

Stratégies de résolution

Pour gérer les limites de longueur :

1. Utilisez des noms de paramètres et des valeurs plus courts si possible
2. Divisez les grandes valeurs de paramètres en plusieurs paramètres plus petits
3. Utilisez des alias de configuration pour mapper des alias courts à des valeurs plus longues (voir [MediaTailor vue d'ensemble des alias de configuration](#))
4. Envisagez d'utiliser un stockage externe pour les données volumineuses avec des références de paramètres

Erreurs d'alias de configuration

Les problèmes d'alias de configuration peuvent entraîner des erreurs HTTP 400 ou des valeurs de paramètres inattendues.

Erreurs d'alias de configuration courantes

Les erreurs suivantes se produisent fréquemment avec les alias de configuration :

- Erreur HTTP 400 : valeur d'alias manquante ou non valide
- Les variables de domaine ne sont pas résolues correctement
- Les paramètres du joueur ne sont pas remplacés par des valeurs d'alias

Liste de contrôle de résolution

Pour résoudre les erreurs d'alias de configuration :

1. Vérifiez que toutes les variables de domaine sont définies comme `ConfigurationAliases`
2. Assurez-vous que les variables des paramètres du joueur utilisent un `player_params.` préfixe
3. Vérifiez que la liste des valeurs aliasées est exhaustive pour les variables de domaine dans les URL critiques
(`VideoContentSourceUrl`, `AdSegmentUrlPrefix`, `ContentSegmentUrlPrefix`)
4. Vérifiez que les demandes d'initialisation de session spécifient des valeurs d'alias valides
5. Valider la structure JSON du `ConfigurationAliases` paramètre

Pour obtenir des conseils de dépannage détaillés, consultez [MediaTailor guide de dépannage des alias de configuration](#).

Exemple Validation des alias de configuration

Assurez-vous que votre configuration inclut tous les alias requis :

```
"ConfigurationAliases": {
  "player_params.origin_domain": {
    "pdx": "abc.mediapackage.us-west-2.amazonaws.com",
    "iad": "xyz.mediapackage.us-east-1.amazonaws.com"
    // Must include all possible values used in session initialization
  }
}
```

Problèmes de flux de traitement des paramètres

Comprendre le flux de traitement des paramètres permet de résoudre les problèmes liés à la transmission et à la transformation des paramètres.

Ordre de traitement des paramètres

MediaTailor traite les paramètres dans l'ordre suivant :

1. Validation des paramètres d'initialisation de session
2. Résolution des alias de configuration (le cas échéant)
3. Filtrage des paramètres (ADS, origine ou manifeste)
4. Encodage et mise en forme des URL
5. Application de paramètres aux URL

Flux de paramètres de débogage

Pour résoudre les problèmes liés au traitement des paramètres :

1. Vérifiez que les paramètres sont correctement spécifiés lors de l'initialisation de la session
2. Vérifiez que les alias de configuration sont résolus aux valeurs attendues
3. Vérifiez que les paramètres apparaissent dans les URL correctes (manifeste, ADS, origine)
4. Valider que le codage de l'URL est correctement appliqué

Exemple Exemple de flux de paramètres

Initialisation de la session :

```
POST master.m3u8
{
  "playerParams": {"origin_domain": "pdx"},
  "manifestParams": {"test": "123"}
}
```

Après la résolution et le traitement des alias :

- Demande d'origine : `https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd`
- URL du manifeste : `/v1/master/.../index.m3u8?aws.sessionId=session&test=123`

Considérations sur la sécurité et bonnes pratiques en la matière

MediaTailor met en œuvre des mesures de sécurité pour la gestion des paramètres afin d'éviter les problèmes de sécurité courants.

Mesures de sécurité

MediaTailor met en œuvre les mesures de sécurité suivantes :

1. Limitations de taille d'entrée pour éviter le gonflement de la base de données
2. Encodage et nettoyage appropriés des entrées de l'utilisateur
3. URL-encoding d'entrées pour éviter la corruption des réponses

Bonnes pratiques

Suivez ces bonnes pratiques pour une gestion sécurisée des paramètres :

- Validez les valeurs des paramètres côté client avant de les envoyer
- Utiliser des alias de configuration pour limiter les valeurs de paramètres possibles
- Évitez d'inclure des informations sensibles dans les paramètres
- Surveillez l'utilisation des paramètres pour détecter les modèles inhabituels
- Maintenez les valeurs des paramètres dans les limites de longueur recommandées

MediaTailor guide de dépannage des alias de configuration

AWS Elemental MediaTailor fournit des conseils de dépannage systématiques pour les problèmes courants liés aux alias de configuration et les scénarios d'erreur.

Erreurs de validation des alias de configuration

Lorsque des alias de configuration sont manquants ou non valides, MediaTailor renvoie des réponses d'erreur spécifiques pour aider à identifier le problème.

Scénarios d'erreur courants

Le tableau suivant décrit les erreurs d'alias de configuration courantes et les étapes à suivre pour les résoudre :

Erreur	Cause	Résolution
HTTP 400 : alias de paramètre de joueur non valide	La valeur du paramètre du joueur est introuvable dans ConfigurationAliases	Vérifiez que la valeur du paramètre du joueur existe en tant que clé dans le ConfigurationAliases mappage correspondant
HTTP 400 : alias de configuration requis manquant	Variable de domaine utilisée sans ConfigurationAliases entrée correspondante	Ajoutez le paramètre de joueur manquant à ConfigurationAliases avec tous les mappages d'alias requis

Erreur	Cause	Résolution
HTTP 400 : échec de la validation de la configuration	ConfigurationAliases la structure est mal formée ou incomplète	Validez la structure JSON et assurez-vous que toutes les variables de domaine ont des alias correspondants
Remplacement de chaînes vides dans les URL	Non-domain alias de variable introuvable	Ajoutez le mappage d'alias manquant ou fournissez une valeur par défaut dans ConfigurationAliases

Liste de contrôle de validation

Utilisez la liste de contrôle suivante pour valider la configuration de vos alias de configuration :

1. Couverture des variables de domaine : assurez-vous que toutes les variables utilisées dans les parties de domaine des URL comportent des ConfigurationAliases entrées correspondantes
2. Exhaustivité des alias : vérifiez que toutes les valeurs possibles des paramètres du joueur sont incluses en tant que clés dans les mappages d'alias
3. Structure JSON : vérifiez que le ConfigurationAliases JSON est correctement formaté et imbriqué
4. Dénomination des paramètres : vérifiez que tous les paramètres du lecteur utilisent le `player_params.` préfixe
5. Cohérence des valeurs : assurez-vous que les valeurs d'alias sont valides pour l'usage auquel elles sont destinées (URL, noms de profil, etc.)

Résolution des alias de configuration de débogage

Suivez cette approche systématique pour résoudre les problèmes de résolution des alias de configuration.

Step-by-step méthodologie de débogage

Procédez comme suit pour identifier et résoudre les problèmes d'alias de configuration :

Procédure de débogage des alias de configuration

1. Vérifiez la structure de configuration : vérifiez que votre configuration de lecture est correctement formatée ConfigurationAliases

```
{
  "ConfigurationAliases": {
    "player_params.example_param": {
      "alias1": "value1",
      "alias2": "value2"
    }
  }
}
```

2. Vérifiez le format des paramètres du lecteur : assurez-vous que l'initialisation de la session inclut des paramètres de lecteur correctement formatés

```
{
  "playerParams": {
    "example_param": "alias1"
  }
}
```

3. Valider le mappage des alias : confirmez que la valeur du paramètre du joueur (« alias1 ») existe en tant que clé dans le mappage ConfigurationAliases
4. Testez avec une configuration simple : commencez par une configuration minimale pour isoler le problème
5. Surveiller les réponses aux erreurs : vérifiez les réponses aux MediaTailor erreurs pour des messages de validation spécifiques
6. Vérifier les URL résolues : confirmez que les URL résolues finales sont valides et accessibles

Meilleures pratiques en matière de configuration des alias

Suivez ces bonnes pratiques pour garantir une mise en œuvre fiable des alias de configuration.

Considérations sur la sécurité

Implémentez les mesures de sécurité suivantes lorsque vous utilisez des alias de configuration :

- Validation des entrées : validez toutes les valeurs des paramètres du joueur avant de les utiliser dans la résolution des alias
- Nettoyage des valeurs d'alias : assurez-vous que les valeurs d'alias contiennent uniquement les caractères et les formats attendus
- Restrictions de domaine : limitez les alias de domaine aux domaines fiables et contrôlés

- **Contrôle d'accès** : Restreindre la modification de la configuration au personnel autorisé uniquement

Optimisation des performances

Optimisez les performances des alias de configuration grâce aux recommandations suivantes :

- **Minimiser le nombre d'alias** : utilisez uniquement les alias nécessaires pour réduire les frais de traitement
- **Dénomination efficace** : utilisez des conventions de dénomination claires et cohérentes pour les alias et les paramètres
- **Valeurs par défaut** : fournissez des alias par défaut raisonnables pour les cas d'utilisation courants
- **Mise en cache de configuration** : tirez parti MediaTailor de la mise en cache de configuration pour améliorer les performances

Maintenance et surveillance

Maintenez des opérations d'alias de configuration fiables grâce aux pratiques suivantes :

- **Validation régulière** : vérifiez périodiquement que tous les mappages d'alias sont actuels et fonctionnels
- **Surveillance des erreurs** : surveillez les erreurs HTTP 400 liées à des alias manquants ou non valides
- **Documentation** : Conservez une documentation claire sur tous les mappages d'alias et leurs objectifs
- **Procédures de test** : mettez en œuvre des tests complets pour toutes les combinaisons d'alias

MediaTailor paramètres de requête manifeste

AWS Elemental MediaTailor gère les paramètres de requête à différentes fins : paramètres de requête manifeste pour le routage et l'autorisation du CDN, et autres paramètres de requête pouvant être utilisés pour des fonctionnalités spécifiques à l'origine.

AWS Elemental MediaTailor préserve les paramètres de requête dès l'initialisation de la session et les ajoute au manifeste personnalisé URLs et à d'autres actifs. Utilisez cette fonctionnalité lorsque vous disposez d'un réseau de diffusion de contenu (CDN) entre MediaTailor et le lecteur client.

Utilisez les paramètres de requête du manifeste lorsque votre CDN a besoin des paramètres de requête pour les raisons suivantes :

- Routage dynamique vers différents points de MediaTailor terminaison
- Autorisation par jeton

Comportement côté client par rapport au CDN

MediaTailor ajoute des paramètres de requête pour les points de terminaison de reporting côté client, mais ne les ajoute pas pour les segments CDN. La fonctionnalité mise à jour fournit une prise en charge plus complète des paramètres de requête sur divers MediaTailor actifs, améliorant ainsi la flexibilité pour le routage des CDN et les cas d'utilisation des autorisations.

MediaTailor ajoute des paramètres de requête pour les points de terminaison de reporting côté client, mais il n'ajoute pas les paramètres de requête pour les segments CloudFront (ou d'autres CDN).

Pour utiliser la préservation des paramètres, contactez [AWS Support](#) pour demander que le transfert des paramètres de requête du manifeste soit activé.

Le comportement varie entre HLS et DASH, et entre l'initialisation de session explicite et implicite. Les rubriques suivantes décrivent comment configurer les demandes d'initialisation de session afin que MediaTailor les paramètres soient transmis au manifeste :

MediaTailor gestion des paramètres de requête pour les origines

AWS Elemental MediaTailor gère les paramètres de requête différemment en fonction de leur objectif. Les paramètres de requête du manifeste (préfixés `parmanifest.`) sont utilisés pour le routage et l'autorisation du CDN, tandis que d'autres paramètres de requête peuvent être utilisés pour des fonctionnalités spécifiques à l'origine.

Affichage décalé dans le temps avec MediaPackage

Pour une fonctionnalité de visualisation décalée dans le temps avec MediaPackage, vous pouvez inclure `start` des `end` paramètres dans vos demandes. Ces paramètres définissent des fenêtres de contenu spécifiques pour le visionnage initial et le visionnage de rattrapage.

Exemple demande de visualisation décalée dans le temps

Incluez les paramètres de début et de fin dans votre demande de manifeste pour un affichage décalé dans le temps :

```
GET /v1/master/111122223333/originId/index.m3u8?  
start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```

Comportement des paramètres pendant les sessions

Les paramètres de requête sont traités lors de l'initialisation de la session. Pour un affichage décalé dans le temps ou pour d'autres fonctionnalités spécifiques à l'origine :

- Initialisation de session : incluez les paramètres requis dans la demande de manifeste initiale
- Persistance des paramètres : les paramètres sont associés à la session et conservés tout au long de la lecture
- Modification des paramètres : pour modifier les fenêtres de décalage horaire ou d'autres paramètres, créez une nouvelle session avec des valeurs mises à jour

Important

La gestion spécifique des paramètres de requête dépend de votre configuration d'origine et des paramètres pris en charge par votre origine de contenu. Pour MediaPackage l'intégration, assurez-vous que votre CDN est configuré pour transmettre les paramètres de requête nécessaires, comme décrit dans [Configuration des paramètres de requête essentiels](#).

MediaTailor restrictions relatives aux caractères des paramètres et codage des URL

AWS Elemental MediaTailor prend en charge des caractères spécifiques dans les paramètres de requête du manifeste. Vous pouvez utiliser le codage URL pour les caractères spéciaux.

Caractères pris en charge avec encodage URL

Les caractères spéciaux suivants sont pris en charge par le codage d'URL :

- période (.) = %2E
- tiret (-) = %2D
- trait de soulignement (_) = %5F
- pourcentage (%) = %25

- tilde (~) = %7E
- barre oblique (/) = %2F
- astérisque (*) = %2A
- est égal à (=) = %3D
- une question (?) = 3 %F

Support du codage d'URL

MediaTailor prend en charge le signe du pourcentage (%) lorsque vous l'utilisez dans le codage d'URL (par exemple, hello%20world = hello world). Vous pouvez utiliser n'importe quel caractère codé en URL, à condition qu'il s'agisse d'encodages d'URL valides conformément à la spécification HTTP.

Important

MediaTailor ne prend pas en charge les caractères doubles tels que %%% ou ==.

Considérations sur la sécurité

MediaTailor met en œuvre les mesures de sécurité suivantes pour la gestion des paramètres :

1. Limitations de taille d'entrée pour éviter le gonflement de la base de données
2. Encodage et nettoyage appropriés des entrées utilisateur
3. Encodage URL de l'entrée pour empêcher la corruption de la réponse

Rubriques

- [MediaTailor Initialisation de session HLS implicite](#)
- [MediaTailor Initialisation implicite de session DASH](#)
- [MediaTailor Initialisation explicite des sessions HLS et DASH](#)
- [MediaTailor comportement des paramètres spécifiques au protocole](#)
- [MediaTailor Intégration du CDN et routage des paramètres](#)

MediaTailor Initialisation de session HLS implicite

AWS Elemental MediaTailor inclut les paramètres de requête dans les liens vers MediaTailor les ressources lorsque votre demande inclut des paramètres de requête avec la clé `manifest.*`.

L'exemple suivant illustre ce format de demande :

```
GET /v1/master/111122223333/originId/index.m3u8?manifest.test=123&other=456
```

Les liens n'incluent pas le `manifest.` préfixe.

Application de paramètres dans HLS

Pour les sessions HLS implicites, MediaTailor applique des paramètres aux emplacements suivants dans la hiérarchie du manifeste :

- Liste de lecture multivariée URLs
- Liste de lecture multimédia URLs
- Segment de contenu URLs
- Segment publicitaire URLs
- Initialisation du HLS URLs

Exemple liste de lecture multivariante

L'exemple suivant montre comment MediaTailor inclure les paramètres de requête dans l'URL de la liste de lecture multivariée :

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI=".../.../
manifest/111122223333/originId/session/1.m3u8?test=123",GROUP-
ID="subtitles",DEFAULT=YES,NAME="caption_1"
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-
BANDWIDTH=2525657,RESOLUTION=960x540,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=2665212
.../.../manifest/111122223333/originId/session/0.m3u8?test=123
```

Exemple playlist multimédia

L'exemple suivant montre comment MediaTailor inclure les paramètres de requête dans URLs les segments de contenu :

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:28716269
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://origin.com/contentSegment_1.ts?originQueryParam=foo&test=123
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
../../../../segment/111122223333/originId/session/0/2?test=123
```

MediaTailor Initialisation implicite de session DASH

AWS Elemental MediaTailor crée une session pour le client et la redirige avec des paramètres de requête lorsque le client fait une demande de manifeste sans session. L'exemple suivant illustre ce format de demande :

```
GET /v1/dash/111122223333/originId/index.mpd?manifest.test=123&other=456
```

MediaTailor crée une session pour le client et la redirige avec les paramètres de requête :

```
/v1/dash/111122223333/originId/index.mpd?sessionId=session&test=123
```

Application des paramètres dans DASH

La réponse au manifeste DASH inclut les paramètres de requête à différents emplacements, notamment les segments de contenu, les segments publicitaires et l'initialisation URLs. MediaTailor applique des paramètres aux éléments suivants :

- Éléments de localisation du manifeste DASH
- SegmentTemplate attributs d'initialisation
- SegmentTemplate attributs des médias
- Segment de contenu URLs
- Segment publicitaire URLs

Lorsque le client fait la demande, il MediaTailor répond avec un manifeste DASH similaire à l'exemple suivant. La première période étant une période de contenu, le paramètre de requête du manifeste MediaTailor n'y est pas inséré. Au cours de la seconde période, qui est une période publicitaire, MediaTailor insère le paramètre de requête du manifeste dans l'initializationattribut et l'attribut de media l'SegmentTemplateélément. L'Locationélément possède également les paramètres de requête du manifeste.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD availabilityStartTime="2018-07-27T09:48:23.634000+00:00"
  id="201" minBufferTime="PT30S" minimumUpdatePeriod="PT15S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011" publishTime="2023-02-14T23:37:43"
  suggestedPresentationDelay="PT25.000S" timeShiftBufferDepth="PT56.997S" type="dynamic"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/ittf/
PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://origin.com/contentSegments/</BaseURL>
  <Location>https://mediatailor.com/v1/dash/11112223333/originId/index.mpd?
test=123&aws.sessionId=session</Location>
  <Period duration="PT29.963S" id="28737823" start="PT143732873.178S">
    <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
  segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
  subsegmentStartsWithSAP="1">
      <Representation bandwidth="2200000" codecs="avc1.640029"
  frameRate="30000/1001" height="540" id="1" width="960">
        <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1611174111" media="index_video_7_0_${Number$.mp4?m=1611174111"
  presentationTimeOffset="4311986195351" startNumber="28737828" timescale="30000">
          <SegmentTimeline>
            <S d="180180" t="4311986911066"/>
            <S d="3003" t="4311987091246"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>
  </Period>
  <Period id="28737829_1" start="PT39925H48M23.141S">
    <BaseURL>https://mediatailor.com/v1/
dashsegment/11112223333/originId/session/28737829/28737829_1/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
  mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
  subsegmentAlignment="true" subsegmentStartsWithSAP="1">
      <SegmentTemplate startNumber="1" timescale="90000"/>
    </AdaptationSet>
  </Period>
</MPD>
```

```

    <Representation bandwidth="2200000" codecs="avc1.64001f" height="540"
id="1" width="960">
      <SegmentTemplate initialization="asset_540_2_0init.mp4?test=123"
media="asset_540_2_0_${Number%09d$.mp4?test=123" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="87087" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
</MPD>

```

MediaTailor Initialisation explicite des sessions HLS et DASH

AWS Elemental MediaTailor inclut les paramètres de requête `manifestParams` as dans la liste de lecture multivariante et le suivi URLs dans la réponse lorsque le client fait une demande explicite d'initialisation de session.

Méthodes d'initialisation de session

Pour une initialisation explicite de session, vous pouvez utiliser POST avec le corps de la requête ou GET avec les paramètres de requête :

1. POST avec le corps de la requête :

```

POST /v1/session/111122223333/originId/index.m3u8
{
  "adsParams": {"param1": "value1", "param2": "value2", "param3": "value3"},
  "manifestParams": {"test": "123"}
}

```

2. GET avec paramètres de requête :

```

GET /v1/session/111122223333/originId/index.m3u8?
ads.param1=value1&ads.param2=value2&manifestParams.test=123

```

Exemple demande d'initialisation de session

```

POST /v1/session/111122223333/originId/index.m3u8

```

```
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2",
    "param3": "value3"
  },
  "manifestParams": {
    "test": "123"
  },
  "reportingMode": "client"
}
```

Exemple manifeste et réponse de suivi

```
{
  "manifestUrl": "/v1/master/111122223333/originId/index.m3u8?
aws.sessionId=session&test=123",
  "trackingUrl": "/v1/tracking/111122223333/originId/session?test=123"
}
```

Les réponses au manifeste pour la session ont le caractère spécifique MediaTailor URLs similaire `manifestParams` aux flux de travail implicites d'initialisation de session décrits précédemment. La principale différence est que les paramètres du manifeste pour l'initialisation explicite d'une session ne commencent `manifest.` pas par.

Les paramètres de requête du manifeste sont immuables et vous ne pouvez les définir que lors de l'initialisation de la session. Si un client fait plusieurs demandes de playlist multivariantes pour une même session, MediaTailor il ne met pas à jour les paramètres de requête du manifeste après la première demande.

Flux de traitement des paramètres

Vous ne pouvez spécifier les paramètres qu'une seule fois, au moment de l'initialisation. Les alias de configuration sont résolus en valeurs réelles avant le transfert. Par exemple : `player_params.ad_type=customized` est résolu en `ad_type=abc12345` fonction de la `ConfigurationAliases` configuration.

MediaTailor comportement des paramètres spécifiques au protocole

AWS Elemental MediaTailor gère les paramètres de requête du manifeste différemment pour les protocoles HLS et DASH. Chaque type de protocole possède des emplacements d'application et des méthodes de traitement spécifiques.

Comparaison entre la gestion des paramètres HLS et DASH

Le tableau suivant compare le mode de gestion MediaTailor des paramètres de requête du manifeste entre les protocoles HLS et DASH :

Aspect	Comportement HLS	Comportement DASH
Application des paramètres	Appliqué directement au manifeste URLs et au segment URLs	Appliqué aux éléments de localisation, SegmentTemplate aux attributs et au segment URLs
Hiérarchie du manifeste	Liste de lecture multivariée → Listes de lecture multimédia → Segments	MPD → Périodes → AdaptationSets → Représentations
Initialisation URLs	Appliqué à HLS init lorsqu'il est présent URLs	Appliqué aux attributs SegmentTemplate d'initialisation
Gestion des sessions	Paramètres préservés lors des mises à jour des playlists	Paramètres inclus dans l'élément de localisation MPD pour la continuité de session
Gestion des segments publicitaires	Appliqué au segment publicitaire URLs dans les playlists multimédias	Appliqué aux attributs SegmentTemplate multimédias de la période d'annonce

Emplacement des applications de paramètres

MediaTailor applique les paramètres de requête du manifeste aux emplacements suivants :

Application des paramètres HLS

Pour les flux HLS, MediaTailor applique les paramètres de requête du manifeste à :

- Liste de lecture multivariante URLs : les paramètres sont ajoutés aux références aux listes de lecture multimédia
- Liste de lecture multimédia URLs : les paramètres sont inclus dans le segment URLs des listes de lecture multimédia
- Segment de contenu URLs : tous les segments de contenu incluent les paramètres de requête du manifeste
- Segment publicitaire URLs : les segments publicitaires reçoivent des paramètres pour le routage et l'autorisation du CDN
- Initialisation HLS URLs : les segments d'initialisation incluent des paramètres lorsqu'ils sont présents dans le flux
- Segment Slate URLs : le contenu de Slate inclut des paramètres pour un comportement CDN cohérent

Exemple Exemple d'application de paramètres HLS

Compte tenu de l'initialisation de la session :

```
GET /v1/master/123456789/originId/index.m3u8?
manifest.auth_token=abc123&manifest.region=us-west
```

La liste de lecture multivariante inclut des paramètres dans les références de listes de lecture multimédia :

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-STREAM-INF:BANDWIDTH=2665212,RESOLUTION=960x540
../../manifest/123456789/originId/session/0.m3u8?auth_token=abc123&region=us-west
```

La liste de lecture multimédia inclut les paramètres du segment URLs :

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXTINF:6.006,
https://origin.com/segment1.ts?auth_token=abc123&region=us-west
#EXTINF:6.006,
../../segment/123456789/originId/session/0/2?auth_token=abc123&region=us-west
```

Application des paramètres DASH

Pour les flux DASH, MediaTailor applique les paramètres de requête du manifeste à :

- Éléments de localisation MPD : l'élément de localisation inclut des paramètres pour les demandes d'actualisation du manifeste
- SegmentTemplate attributs d'initialisation : le segment URLs d'initialisation inclut des paramètres
- SegmentTemplate attributs multimédia : les modèles d'URL de segment multimédia incluent des paramètres
- Segment de contenu URLs : tous les segments de contenu générés à partir de modèles incluent des paramètres
- Segment publicitaire URLs : les segments de période publicitaire incluent des paramètres pour l'intégration du CDN
- Redirections de rapports côté serveur : 302 redirections vers des segments publicitaires préservent les paramètres

Exemple Exemple d'application des paramètres DASH

Compte tenu de l'initialisation de la session :

```
GET /v1/dash/123456789/originId/index.mpd?  
manifest.auth_token=abc123&manifest.region=us-west
```

Le manifeste DASH inclut des paramètres situés à plusieurs endroits :

```
<MPD>  
  <Location>https://mediatailor.com/v1/dash/123456789/originId/index.mpd?  
  auth_token=abc123&region=us-west&aws.sessionId=session</Location>  
  <Period>  
    <AdaptationSet>  
      <Representation>  
        <SegmentTemplate  
          initialization="init.mp4?auth_token=abc123&region=us-west"  
          media="segment_${Number$.mp4?auth_token=abc123&region=us-west"/>  
        </Representation>  
      </AdaptationSet>  
    </Period>  
</MPD>
```


MediaTailor Intégration du CDN et routage des paramètres

AWS Elemental MediaTailor les paramètres de requête manifeste permettent des scénarios d'intégration de CDN sophistiqués. Vous pouvez les utiliser pour le routage dynamique, l'autorisation et l'équilibrage de charge.

Cas d'utilisation du routage CDN

Les scénarios courants d'intégration du CDN qui tirent parti des paramètres de requête du manifeste sont les suivants :

- Routage géographique : acheminer les demandes vers des MediaTailor points de terminaison spécifiques à une région en fonction de l'emplacement du spectateur
- Autorisation basée sur des jetons : transmettez des jetons d'autorisation via le CDN MediaTailor pour un accès sécurisé au contenu
- Équilibrage de charge : répartissez le trafic sur plusieurs MediaTailor points de terminaison à l'aide de la logique de routage du CDN
- Tests A/B : acheminez différents segments d'utilisateurs vers différentes MediaTailor configurations à des fins de test
- Optimisation spécifique à l'appareil : acheminer les demandes en fonction du type ou des capacités de l'appareil

Conservation des paramètres entre les couches CDN

MediaTailor garantit que les paramètres de requête du manifeste sont préservés sur plusieurs couches CDN et types de demandes :

1. Demande initiale : les paramètres sont extraits de la demande d'initialisation de session
2. Génération du manifeste : les paramètres sont appliqués à tous les URLs éléments pertinents du manifeste
3. Demandes de segment : les paramètres sont inclus dans tous les segments URLs pour un comportement CDN cohérent
4. Insertion d'annonces : les paramètres sont préservés lors de l'insertion de publicités et du remplacement des segments

Exemple Flux d'autorisation CDN

L'exemple suivant illustre un flux d'autorisation CDN complet à l'aide des paramètres de requête du manifeste :

1. Le client demande un manifeste avec un jeton d'autorisation :

```
GET https://cdn.example.com/mediatailor/v1/master/123456789/originId/index.m3u8?
manifest.auth_token=jwt_token_here&manifest.user_id=12345
```

2. Le CDN transmet la demande à MediaTailor avec les paramètres suivants :

```
GET https://mediatailor.amazonaws.com/v1/master/123456789/originId/index.m3u8?
manifest.auth_token=jwt_token_here&manifest.user_id=12345
```

3. MediaTailor génère un manifeste avec des paramètres appliqués à tous URLs :

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=2665212
../.../manifest/123456789/originId/session/0.m3u8?
auth_token=jwt_token_here&user_id=12345
```

4. Les demandes de segment suivantes incluent les paramètres d'autorisation du CDN :

```
GET https://cdn.example.com/mediatailor/segment/123456789/originId/session/0/1?
auth_token=jwt_token_here&user_id=12345
```

Rapports et données de suivi

MediaTailor propose deux options permettant de suivre la quantité de publicité regardée par un internaute et de générer des rapports à ce sujet. Dans l'approche de reporting publicitaire côté serveur, MediaTailor suit la publicité et envoie des balises (signaux de suivi) directement au serveur publicitaire. Sinon, dans l'approche de suivi côté client, le lecteur client (l'appareil de l'utilisateur) suit l'annonce et envoie les balises au serveur publicitaire. Le type de rapport publicitaire utilisé lors d'une session de lecture dépend de la demande spécifique du joueur pour lancer la session MediaTailor.

Pour plus d'informations sur la transmission des données de session et de joueur au serveur publicitaire à l'aide de variables dynamiques, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour plus de détails sur les paramètres d'initialisation de session, consultez [MediaTailor paramètres de requête manifeste](#).

Rubriques

- [MediaTailor suivi des publicités et rapports côté serveur](#)
- [Client-side suivi des publicités](#)

MediaTailor suivi des publicités et rapports côté serveur

AWS Elemental MediaTailor utilise par défaut les rapports côté serveur pour un suivi et une mesure complets des publicités. Avec le reporting côté serveur, lorsque le lecteur demande une URL de publicité du manifeste, le service indique la consommation de publicité directement à l'URL de suivi publicitaire. Une fois que le lecteur a initialisé une session de lecture avec MediaTailor, aucune autre saisie n'est requise de votre part ou de votre part pour effectuer des rapports côté serveur. Au fur et à mesure de la lecture de chaque annonce, MediaTailor envoie des balises au serveur publicitaire pour indiquer le nombre de vues de l'annonce. MediaTailor envoie des balises pour le début de l'annonce et pour la progression de l'annonce en quartiles : le premier quartile, le point médian, le troisième quartile et la fin de l'annonce.

Pour effectuer le reporting publicitaire côté serveur

- Depuis le lecteur, initialisez une nouvelle session de MediaTailor lecture à l'aide d'une requête dans l'un des formats suivants, conformément à votre protocole :
 - Exemple : format HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?ads.<key-value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

- Exemple : format DASH

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?ads.<key-value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

Les paires clé-valeur sont les paramètres de ciblage dynamique pour le suivi publicitaire.

Pour plus d'informations sur l'ajout de paramètres à la demande, consultez [the section called "Variables publicitaires dynamiques"](#).

AWS Elemental MediaTailor répond à la demande avec l'URL du manifeste. Le manifeste contient les URL des manifestes des médias. Les manifestes des médias contiennent des liens intégrés pour les demandes de segments publicitaires.

 Note

Lorsqu' MediaTailor une double barre oblique (//) apparaît dans une URL de suivi, les barres obliques sont réduites à une (/).

Lorsque le lecteur demande une lecture à partir d'une URL de segment publicitaire (chemin /v1/segment), AWS Elemental MediaTailor envoie la balise appropriée au serveur publicitaire via les URL de suivi publicitaire. Dans le même temps, le service émet une redirection vers le segment publicitaire réel. Le segment publicitaire se trouve soit dans la CloudFront distribution Amazon où MediaTailor les publicités transcodées sont stockées, soit dans le réseau de diffusion de contenu (CDN) sur lequel vous avez mis l'annonce en cache.

Les sections suivantes fournissent des informations supplémentaires sur l'utilisation du suivi des publicités côté serveur à partir de. MediaTailor

Rubriques

- [Server-side suivi avec insertion d'annonces guidée par serveur \(SGAI\)](#)
- [Server-side glossaire des balises de suivi](#)
- [Server-side suivi du calendrier et du comportement de mise en cache](#)
- [MediaTailor fonctionnalités et capacités de suivi côté serveur](#)

Server-side suivi avec insertion d'annonces guidée par serveur (SGAI)

Lorsque vous utilisez l'insertion d'annonces guidée par le serveur (SGAI), le suivi côté serveur utilise un mécanisme de balisage sans session qui diffère de l'approche en mode point décrite ci-dessus. Au lieu d'intégrer MediaTailor des segments d'annonces dans le manifeste de contenu (où il suit les /v1/segment demandes), SGAi renvoie les références des annonces sous forme de listes de lecture distinctes dans une réponse à une liste de ressources avec des métadonnées de balise intégrées dans les URI des annonces.

Comment fonctionne le beaconing côté serveur sans session

Les étapes suivantes décrivent le fonctionnement du beaconing côté serveur pour les sessions SGAI :

1. Initialisation de la session : le joueur demande la liste de lecture multivariante HLS avec. `aws.insertionMode=GUIDED` Server-side le reporting est la valeur par défaut (aucun `aws.reportingMode` paramètre n'est nécessaire). Contrairement au mode Stitched, la réponse d'initialisation de session n'inclut pas de `trackingUrl`
2. Manifeste pouvant être mis en cache : MediaTailor renvoie un manifeste pouvant être mis en cache contenant des EXT-X-DATERANGE balises `CLASS="com.apple.hls.interstitial"` et des X-ASSET-LIST attributs pointant vers le point de terminaison de la liste d'actifs MediaTailor interstitiels.
3. Liste des actifs avec les métadonnées des balises : lorsque le joueur rencontre une pause publicitaire, il récupère la liste des actifs. MediaTailor renvoie une réponse JSON dans laquelle chaque URI d'annonce inclut des métadonnées de balise cryptées :

```
{
  "ASSETS": [
    {
      "DURATION": 30.0,
      "URI": "https://cdn.example.com/ad/master.m3u8?
awsBeaconData=<encrypted>&awsBeaconDomain=<MediaTailor-
endpoint>&awsConfigurationName=<config-name>"
    }
  ]
}
```

Lorsque les rapports côté serveur sont actifs, la réponse ne comporte aucune section. TRACKING Les URI publicitaires contiennent toutes les données des balises.

4. Substitution de variables HLS : le lecteur récupère la playlist multivariante publicitaire. Le manifeste publicitaire utilise des `#EXT-X-DEFINE:QUERYPARAM` directives pour transmettre les paramètres de balise de la chaîne de requête URI aux URL des segments via la substitution de variables HLS :

```
#EXTM3U
#EXT-X-DEFINE:QUERYPARAM="awsBeaconData"
#EXT-X-DEFINE:QUERYPARAM="awsBeaconDomain"
#EXT-X-DEFINE:QUERYPARAM="awsConfigurationName"
```

```
#EXTINF:5.0,  
{$awsBeaconDomain}/segment/hash/{$awsConfigurationName}/{$awsBeaconData}/0/0?  
aws.segmentRelativePath=asset_00001.ts
```

Le lecteur résout les {\$awsConfigurationName} variables {\$awsBeaconData}{awsBeaconDomain}, et à l'aide des valeurs de la chaîne de requête URI du manifeste publicitaire, puis demande chaque segment d'annonce via MediaTailor.

5. La balise se déclenche à la demande de segment : lorsque le joueur demande chaque segment d'annonce, la demande est acheminée MediaTailor. Le service déchiffre les données de la balise, détermine la position du segment dans l'annonce (impression, premier quartile, point médian, troisième quartile ou complet) et envoie la balise de suivi VAST appropriée au serveur publicitaire. MediaTailor redirige ensuite le lecteur vers le segment de contenu publicitaire proprement dit.

Exigences des joueurs pour le beaconing côté serveur SGAI

Pour utiliser le beaconing côté serveur avec SGAI, votre joueur doit répondre aux exigences suivantes :

- HLS version 11 ou ultérieure
- Prise en charge de l'CLASSattribut EXT-X-DATERANGE with pour les interstitiels HLS
- Support pour la substitution de #EXT-X-DEFINE : QUERYPARAM variables (RFC 8216bis). Le joueur doit décoder en pourcentage les valeurs des paramètres de requête avant de les remplacer dans les URL des segments.

Note

Le beaconing SGAI côté serveur est actuellement pris en charge pour HLS uniquement. DASH n'est pas encore pris en charge pour le balisage SGAI côté serveur.

Comparaison avec le suivi côté serveur en mode cousu

Le tableau suivant récapitule les différences entre le suivi côté serveur et l'insertion publicitaire guidée par le serveur :

Aspect	Stitched (SSAI)	Server-guided (SYGAI)
Possibilité de mise en cache manifeste	Per-session, ne peut pas être mis en cache	Peut être mis en cache, partagé entre les spectateurs
Routage des segments publicitaires	/v1/segment/ En utilisant l'ID de session	/v1/segment/ En utilisant un blob de données de balise crypté
État de session pour les balises	Stocké par session dans MediaTailor	Sessionless : tous les états sont enregistrés dans le paramètre chiffré <code>awsBeaconData</code>
URL de suivi au démarrage de la session	Renvoyé dans la réponse d'initialisation de la session	Non fourni : les données des balises sont intégrées dans les URI des annonces dans chaque réponse à la liste d'actifs
Prise en charge de DASH	Pris en charge	Pas encore pris en charge

Note

Pour les sessions SYGAI en direct, vous pouvez activer la prélecture des annonces basée sur les manifestes à l'aide de `aws.guidedPrefetchMode=MANIFEST`. Ceci est distinct de l'API de prélecture basée sur le calendrier utilisée avec les sessions Stitched (SSAI). Pour en savoir plus, consultez [Prélecture guidée avec battements de cœur manifestes](#).

Server-side glossaire des balises de suivi

MediaTailor le suivi côté serveur utilise un ensemble standardisé de balises pour signaler la progression de l'affichage des publicités aux serveurs publicitaires et aux services de vérification. Ces balises sont conformes aux normes de l'Interactive Advertising Bureau (IAB) en matière de mesure des publicités vidéo et fournissent des rapports précis sur les impressions publicitaires et les taux de réalisation.

Server-side types de balises de suivi

Type de balise	En cas de licenciement	Objectif	Détails du chronométrage
Impression	Quand le joueur demande le premier segment publicitaire	Indique que le contenu de l'annonce a commencé à se charger et qu'il est sur le point d'être affiché au spectateur	Annulé dès la première / v1/segment demande d'annonce. Conforme aux directives de l'IAB qui exigent que le contenu publicitaire commence à se charger avant de compter une impression. Voir Server-side flux de travail de suivi la séquence complète.
Démarrer	Quand le lecteur commence à afficher le contenu de l'annonce	Confirme que la lecture de la publicité a bien commencé	Généralement déclenché en même temps que la balise d'impression lors de la première demande de segment, mais représente le début réel du rendu de l'annonce. Cette distinction est importante pour les services de vérification qui suivent séparément les événements d'impression et de démarrage.
Premier quartile	Lorsque le joueur atteint 25 % de la durée de l'annonce	Mesure le visionnage continu des publicités pendant le premier trimestre de la publicité	Déclenché lorsque le joueur demande le segment qui contient le point de 25 % de la durée de l'annonce. Par exemple, dans une annonce de 20 secondes avec des segments de 2 secondes, cela se déclenche généralement à la demande du troisième segment (environ 4 à 6

Type de balise	En cas de licenciement	Objectif	Détails du chronométrage
			secondes après le début de l'annonce).
Point médian	Lorsque le joueur atteint 50 % de la durée de l'annonce	Mesure le visionnage continu des publicités pendant la moitié de la durée de l'annonce	Déclenché lorsque le joueur demande le segment qui contient le point de 50 % de la durée de l'annonce. Par exemple, dans une annonce de 20 secondes avec des segments de 2 secondes, cela se déclenche généralement à la demande du 5e segment (environ 8 à 10 secondes après le début de l'annonce).
Troisième quartile	Lorsque le joueur atteint 75 % de la durée de l'annonce	Mesure le visionnage continu des publicités pendant les trois quarts de la durée de la publicité	Déclenché lorsque le joueur demande le segment qui contient le point de 75 % de la durée de l'annonce. Par exemple, dans une annonce de 20 secondes avec des segments de 2 secondes, cela se déclenche généralement à la demande du 8e segment (entre 14 et 16 secondes environ après le début de l'annonce).

Type de balise	En cas de licenciement	Objectif	Détails du chronométrage
Achevé	Quand le joueur atteint la fin de l'annonce	Confirme que l'intégralité de l'annonce a été transmise au spectateur	Déclenché lorsque le joueur demande le dernier segment de l'annonce. Cela indique que le spectateur a potentiellement vu l'intégralité du contenu de l'annonce. Par exemple, dans une annonce de 20 secondes avec des segments de 2 secondes, cela se déclenche généralement à la demande du 10e segment (environ 18 à 20 secondes après le début de l'annonce).

Note

Le moment exact du déclenchement des balises dépend de la durée du segment et de la durée de l'annonce. MediaTailor calcule la demande de segment appropriée qui correspond à chaque position de quartile en fonction de la durée de l'annonce et de la structure du segment spécifiques.

Server-side suivi du calendrier et du comportement de mise en cache

Dans les rapports côté serveur, les événements de suivi MediaTailor sont déclenchés en fonction des demandes de segments réelles du joueur, et non en fonction des activités d'analyse ou de préchargement du manifeste. Cette approche garantit un comptage précis des impressions, conforme aux normes du secteur en matière de mesure des publicités vidéo.

Principaux principes de chronométrage

MediaTailor le suivi côté serveur suit ces principes de synchronisation fondamentaux :

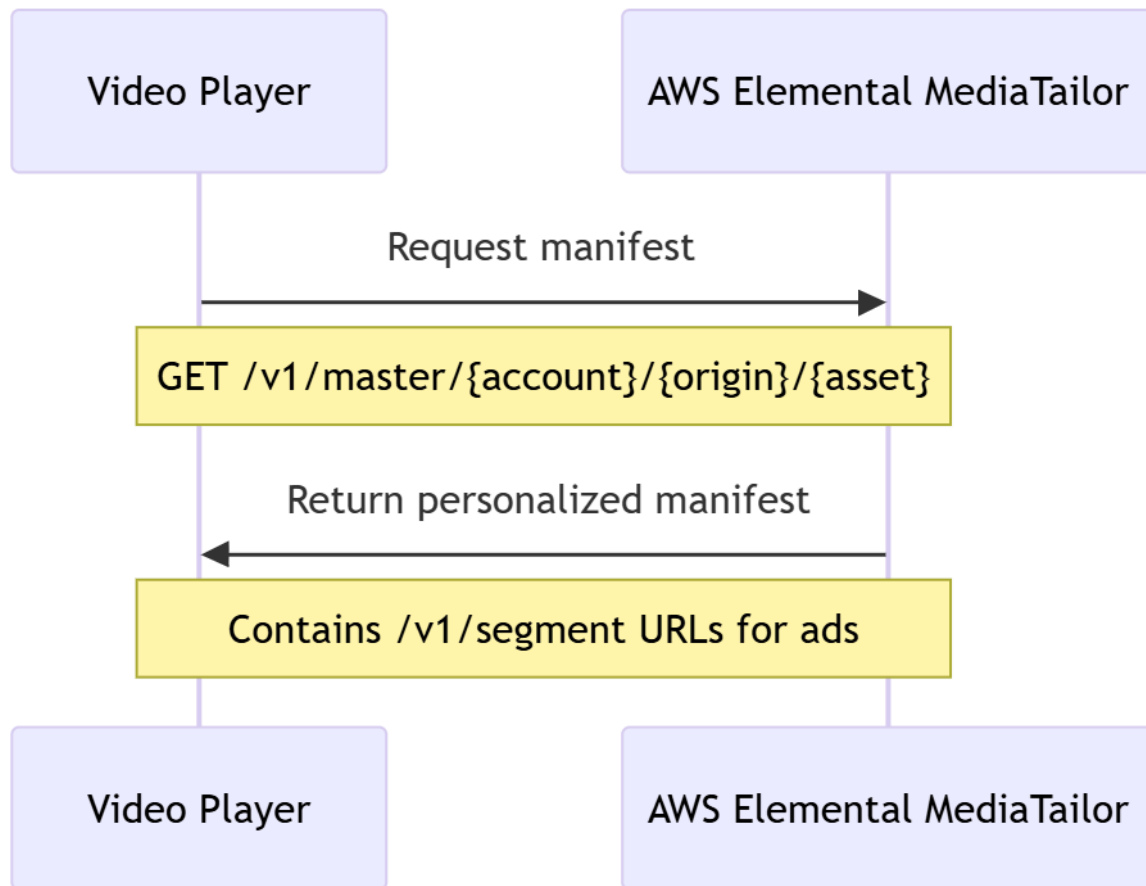
- Les événements de suivi se déclenchent sur les demandes de segments réelles. Les balises ne sont envoyées que lorsque le lecteur envoie des requêtes HTTP à des /v1/segment URL, et non lors de l'analyse ou de la mise en cache du manifeste.
- La mise en cache des joueurs et le préchargement des manifestes ne déclenchent PAS d'événements. Les joueurs peuvent analyser, mettre en cache ou précharger les informations des manifestes sans générer d'événements de suivi.
- La prélecture des segments déclenchera des événements : si les joueurs préextraient des segments publicitaires réels avant leur lecture, cela suit le comportement standard du secteur selon lequel les demandes de segments constituent des impressions valides.
- Chaque v1/segment demande déclenche une balise appropriée - L'événement de suivi spécifique (impression, quartile, achèvement) est déterminé par la position de l'annonce et le segment demandé.
- Le calendrier est conforme aux normes de l'IAB - L'approche suit les directives de l'Interactive Advertising Bureau en matière de mesure des publicités vidéo et de comptage des impressions.

Server-side flux de travail de suivi

Les diagrammes suivants illustrent l'ensemble du flux de travail de suivi côté serveur, indiquant quand les événements de suivi sont déclenchés en relation avec les demandes des joueurs :

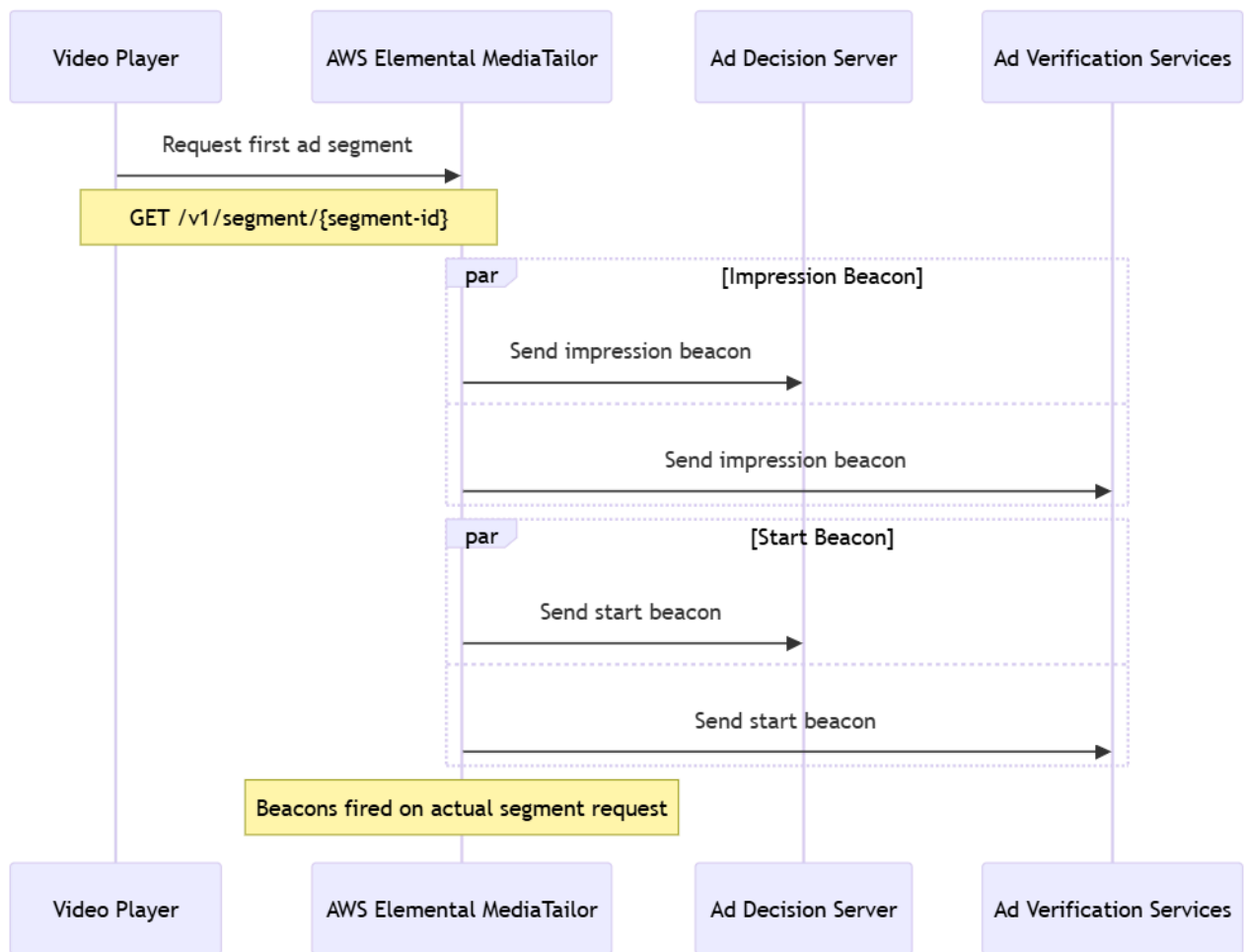
Phase 1 : initialisation de la session

Le joueur demande un manifeste à MediaTailor, qui renvoie un manifeste personnalisé contenant les URL des segments d'annonces :



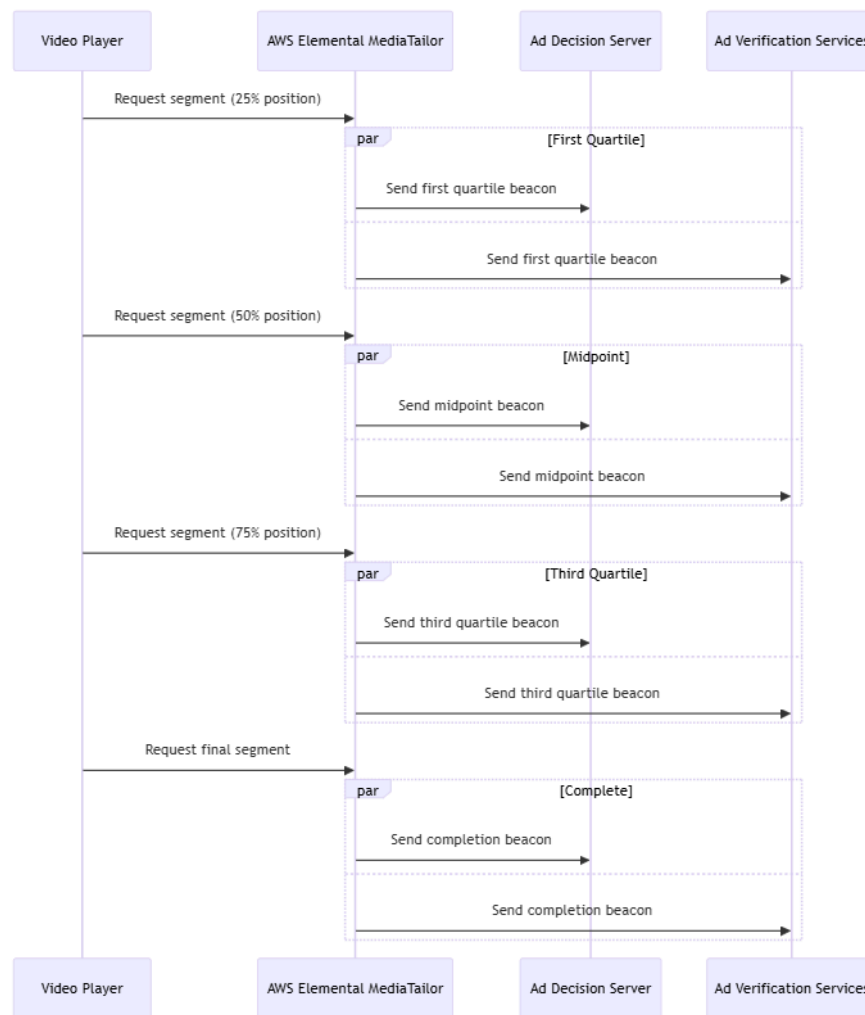
Phase 2 : suivi des demandes d'annonces et des impressions

Lorsque le lecteur demande le premier segment publicitaire, envoie des balises MediaTailor d'impression et de démarrage à la fois au serveur de décision publicitaire et aux services de vérification des publicités :



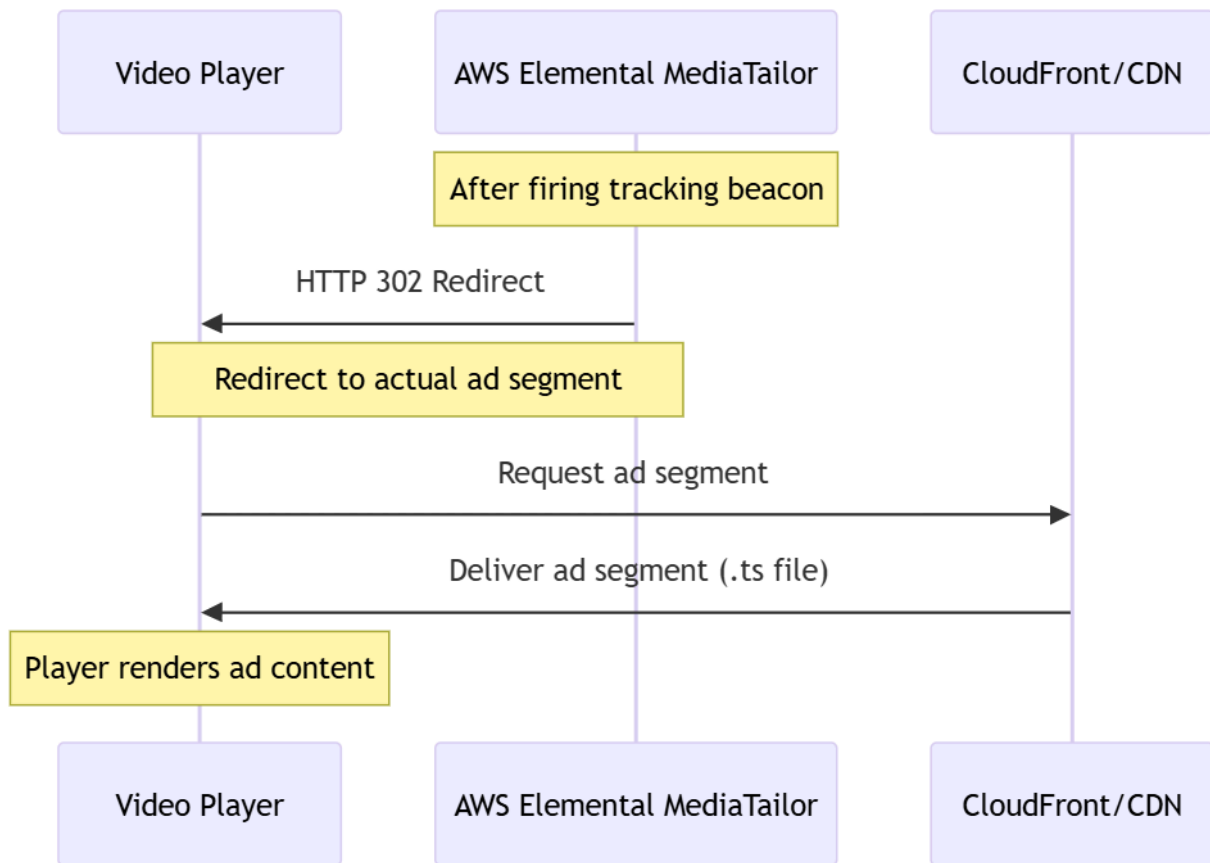
Phase 3 : suivi des quartiles

MediaTailor déclenche des balises de quartile (premier quartile, point médian, troisième quartile, achèvement) en fonction des demandes de segments suivantes :



Phase 4 : Livraison du segment

Après avoir lancé des balises de suivi, MediaTailor redirection vers le segment publicitaire réel depuis Amazon CloudFront ou votre CDN :



Le flux de travail de suivi côté serveur inclut les principaux comportements de synchronisation suivants :

1. Initialisation de la session - Le joueur demande un manifeste à MediaTailor. MediaTailor renvoie un manifeste personnalisé contenant les URL des segments d'annonces avec le `/v1/segment` chemin.
2. Analyse et mise en cache du manifeste : le joueur analyse le manifeste et peut précharger ou mettre en cache les informations des segments. Aucun événement de suivi n'est déclenché pendant cette phase, quel que soit le comportement du joueur en matière de mise en cache.
3. Suivi des demandes de segments publicitaires et des impressions : lorsque le lecteur demande réellement le premier segment publicitaire (généralement pour la lecture), MediaTailor déclenche la balise d'impression et lance le suivi de l'événement vers le serveur de décision publicitaire et

les services de vérification des publicités. Cela se produit lors de la requête HTTP proprement dite adressée à `/v1/segmentURL`, et non lors de l'analyse du manifeste.

4. Suivi des quartiles basé sur les demandes de segments : envoie MediaTailor des balises de quartile (premier quartile, point médian, troisième quartile, achèvement) à la fois au serveur de décision publicitaire et aux services de vérification des publicités en fonction des demandes de segments suivantes qui correspondent aux positions de quartile calculées pendant la durée de l'annonce.
5. Diffusion des segments : après avoir activé la balise de suivi appropriée, MediaTailor émet une redirection HTTP vers le segment publicitaire réel (depuis Amazon CloudFront ou votre CDN).

Considérations relatives à la mise en cache et au préchargement des joueurs

MediaTailor le suivi côté serveur est conçu pour être compatible avec les différentes stratégies de mise en cache et de préchargement des joueurs tout en maintenant une mesure précise des impressions :

- Préchargement du manifeste : les joueurs qui préchargent ou mettent en cache les informations du manifeste ne déclenchent pas d'événements de suivi. Les événements de suivi ne sont déclenchés que lorsque des demandes de segments réelles sont effectuées.
- Prélecture des segments : si un lecteur extrait des segments d'annonces avant leur lecture, des événements de suivi se déclenchent lorsque ces segments sont demandés, éventuellement avant la durée de lecture réelle. Ce comportement est conforme aux normes du secteur qui considèrent les demandes de segments comme des impressions valides.
- Mise en mémoire tampon du lecteur : le comportement standard de mise en mémoire tampon du joueur (demande de segments légèrement avant la lecture) déclenche des événements de suivi aux moments appropriés en fonction du modèle de demande de segment.

Résolution des problèmes de suivi

Si vous constatez des différences entre le suivi MediaTailor côté serveur et les indicateurs tiers, tenez compte des facteurs suivants :

- Différences de comportement entre les joueurs : les stratégies de pré-charge et de mise en mémoire tampon peuvent varier d'un joueur à l'autre, ce qui influe sur le moment où les demandes de segments sont effectuées.

- **État du réseau** - Les mauvaises conditions du réseau peuvent amener les joueurs à demander des segments plusieurs fois ou à des intervalles différents de ceux attendus.
- **Configuration du CDN** : une mise en cache CDN incorrecte des `/v1/segment` demandes peut entraîner des événements de suivi manqués ou dupliqués.
- **Gestion des sessions** : assurez-vous que chaque session de lecture utilise un identifiant de session unique pour éviter les conflits liés aux événements de suivi.

Pour obtenir des conseils de dépannage détaillés, consultez [Dépannage des problèmes courants](#).

MediaTailor fonctionnalités et capacités de suivi côté serveur

AWS Elemental MediaTailor applique automatiquement ces fonctionnalités de suivi intégrées côté serveur afin d'optimiser la précision et la fiabilité des mesures publicitaires. Le système empêche les balises dupliquées, gère le trafic pendant les périodes de volume élevé, assure un séquençage approprié des événements et fournit une surveillance complète des performances sans nécessiter de configuration de votre part. Il vous suffit de vous assurer que votre serveur de décision publicitaire (ADS) fournit les balises de suivi dans la réponse VAST.

Note

Ces fonctionnalités sont disponibles pour les nouveaux clients à compter du 30 septembre 2025. Les clients existants y auront accès tout au long de 2025 dans le cadre de l'amélioration continue des services. Si vous souhaitez accéder immédiatement à ces fonctionnalités, contactez [AWS Support](#).

Note

Ces fonctionnalités s'appliquent à la fois aux méthodes d'insertion d'annonces assemblées (SSAI) et guidées par serveur (SGAI). Les types de balises et la synchronisation sont les mêmes dans les deux modes. Leur mode de MediaTailor déclenchement des balises diffère. [Server-side suivi avec insertion d'annonces guidée par serveur \(SGAI\)](#) Pour plus de détails sur le beaconing côté serveur SGAI, reportez-vous à la section.

Déduplication des balises

MediaTailor empêche le déclenchement de balises dupliquées pour des événements publicitaires identiques. Le système de suivi côté serveur envoie chaque impression, chaque quartile et chaque balise d'achèvement une seule fois par session de visionnage d'une annonce. Lorsque les lecteurs vidéo demandent le même segment publicitaire à plusieurs reprises en raison de l'état du réseau, de modifications du débit ou de stratégies de mise en mémoire tampon, ils suivent les balises déclenchées MediaTailor et bloquent les transmissions redondantes.

La déduplication résout automatiquement les scénarios courants qui entraînent une augmentation du nombre de balises :

- Streaming à débit adaptatif : lorsque les joueurs téléchargent différentes variantes de qualité d'un même segment publicitaire
- Scénarios de nouvelle tentative sur le réseau : lorsque les joueurs demandent à nouveau des segments en raison de problèmes de réseau ou de délais
- Stratégies de mise en mémoire tampon des joueurs : lorsque les joueurs préchargent ou rechargent des segments à des fins de mise en mémoire tampon

Le système est conçu pour déclencher les balises d'impression une seule fois, même lorsque les joueurs passent d'un niveau de qualité à l'autre.

Régulation adaptative et nouvelles tentatives de balises

MediaTailor gère automatiquement les taux de trafic des balises en fonction des indicateurs de réponse du serveur. Le système surveille les modèles de réponse HTTP, les délais de connexion et les codes d'erreur pour détecter la congestion, puis ajuste les débits de trafic en conséquence. Lorsque le système identifie des indicateurs de stress du serveur, il réduit les taux de trafic pour le domaine concerné et augmente automatiquement les taux lorsque les serveurs présentent une capacité améliorée.

Le système surveille l'état du serveur à l'aide des indicateurs suivants :

- Délais de connexion HTTP : lorsque les plateformes de mesure ne répondent pas dans les délais prévus
- Codes de réponse d'erreur : réponses 503, 504 et 507 indiquant une surcharge du serveur. Votre serveur publicitaire doit également prendre en charge ces codes d'erreur pour une compatibilité totale.

- Modèles de réponse - Changements de performance de la plateforme de mesure indiquant des problèmes de capacité

Le comportement de nouvelle tentative tente automatiquement la livraison pendant une heure maximum, avec un délai minimum de 30 secondes entre les tentatives. Ce comportement de nouvelle tentative ne peut pas être configuré.

Gestion du trafic des balises par seconde

Vous pouvez définir des limites TPS pour contrôler les taux de diffusion des balises. Il s'agit du seul paramètre configurable pour les fonctionnalités de suivi côté serveur. Account-level les limites limitent le nombre total de demandes de suivi des publicités envoyées à tous les partenaires de mesure. MediaTailor applique une limite TPS minimale de 10 000 afin de fournir une capacité suffisante pour les opérations à l'échelle de l'entreprise.

Soumettez un ticket d' AWS assistance pour établir les limites du TPS avec les informations suivantes :

- ID de compte AWS : identifiant de votre compte spécifique
- Région cible : région AWS dans laquelle vous souhaitez appliquer la limite TPS
- Seuil TPS souhaité : limite requise pour le nombre de transactions par seconde (minimum 10 000)

Par défaut, il n'y a pas de limite de TPS. Vous pouvez demander une limite de TPS si votre serveur de décision publicitaire (ADS) l'exige, mais la limite doit être supérieure à 10 000 TPS. MediaTailor ne dépassera pas la limite que vous avez spécifiée, mais ne garantit pas un débit constant jusqu'à cette limite. Votre serveur de décision publicitaire vous indiquera les limites de TPS qu'il peut prendre en charge.

In-order balisage

MediaTailor assure automatiquement la diffusion séquentielle des événements de suivi des publicités. Le système préserve l'ordre des balises même en cas de problèmes réseau, de nouvelles tentatives ou de gestion du trafic. Cela garantit que les partenaires de mesure reçoivent les événements dans le bon ordre pour des analyses précises.

Le système suit la séquence de balises standard de l'industrie :

1. Démarrer des événements : se déclenche lorsque la diffusion de la publicité commence

2. Événements du premier quartile - Incendie à 25 % des publicités terminées
3. Événements à mi-parcours - Incendie à 50 % de complétion de la publicité
4. Événements du troisième quartile - Incendie à 75 % des annonces terminées
5. Événements de fin - Se déclenche à la fin des publicités

Ces fonctionnalités fonctionnent automatiquement ensemble :

- Les balises sont maintenues pendant l'étranglement pour maintenir le bon ordre
- Chaque domaine partenaire de mesure possède des files d'événements distinctes pour éviter toute interruption lors des ajustements de taux
- La déduplication permet de suivre le type d'événement et la position dans la chronologie tout en maintenant l'ordre chronologique

Client-side suivi des publicités

À l'aide de l'API de suivi AWS Elemental MediaTailor côté client, vous pouvez intégrer les commandes des joueurs pendant les pauses publicitaires dans les flux de diffusion. Dans le cadre du suivi côté client, le joueur ou le client transmet des événements de suivi, tels que le balisage des impressions et des quartiles, à l'Ad Decision Server (ADS) et à d'autres entités de vérification des publicités. Ces événements permettent de suivre à la fois l'état général des interruptions publicitaires et la disponibilité des publicités individuelles au cours de chaque interruption. Pour plus d'informations sur l'impression et le quartile (ADS) et d'autres entités de vérification des publicités. Pour plus d'informations sur le balisage publicitaire par impression et quartile, consultez [Client-side balisage](#) Pour plus d'informations sur ADS et les autres entités de vérification des publicités, consultez [Client-side intégrations de suivi des publicités](#).

Pour plus d'informations sur la transmission des paramètres des joueurs et des données de session à l'ADS pour le suivi côté client, consultez [MediaTailor variables de lecteur pour les requêtes ADS](#) et [MediaTailor variables de session pour les requêtes ADS](#)

Client-side le suivi permet des fonctionnalités telles que les suivantes :

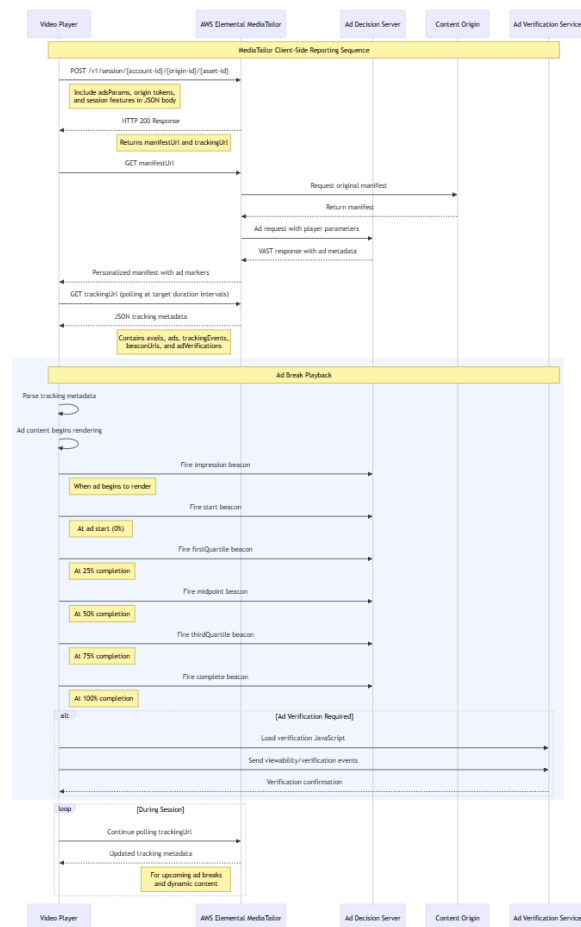
- Ad-break comptes à rebours - Pour plus d'informations, consultez [Compte à rebours publicitaire](#).
- Cliquez sur l'annonce : pour plus d'informations, consultez. [Cliquez sur une annonce](#)
- Affichage d'annonces complémentaires - Pour plus d'informations, consultez [Annonces complémentaires](#).

- Annonces pouvant être ignorées - Pour plus d'informations, consultez [Annonces pouvant être ignorées](#)
- Affichage des icônes VAST pour garantir le respect de la confidentialité - Pour plus d'informations, consultez [Icônes pour Google Why This Ad \(WTA\)](#).
- Contrôle du nettoyage des joueurs pendant les publicités - Pour plus d'informations, consultez [Gommage](#).

À l'aide de l'API de suivi MediaTailor côté client, vous pouvez envoyer des métadonnées au périphérique de lecture qui active des fonctionnalités en plus du suivi côté client :

Client-side flux de travail de reporting

Le schéma suivant montre l'ensemble du flux de production de rapports côté client, de l'initialisation de la session à la lecture des publicités et au balisage :



Le flux de production de rapports côté client comprend les étapes suivantes :

1. **Initialisation de session** : le lecteur vidéo envoie une requête POST au point de terminaison de la MediaTailor session avec des métadonnées JSONadsParams, notamment des jetons d'origine et des fonctionnalités de session. MediaTailor répond `trackingUrl` pendant `manifestUrl` et pour la session.
2. **Demande de manifeste et décision publicitaire** - Le joueur demande le manifeste personnalisé auprès de MediaTailor. MediaTailor demande le manifeste de contenu original depuis l'origine, envoie une demande d'annonce au serveur ADS (Ad Decision Server) en utilisant les paramètres du lecteur, reçoit une réponse VAST avec les métadonnées de l'annonce et fournit un manifeste personnalisé avec des marqueurs publicitaires au lecteur.
3. **Récupération des données de suivi** - Le joueur interroge l'URL de suivi à intervalles réguliers (en fonction de la durée cible pour HLS ou de la période de mise à jour minimale pour DASH). MediaTailor renvoie des métadonnées de suivi JSON contenant les disponibilités, les publicités, les événements de suivi, les URL des balises et les données de vérification des publicités.
4. **Lecture et balisage des publicités** : pendant les pauses publicitaires, le lecteur analyse les métadonnées de suivi, déclenche des balises d'impression lorsque les publicités commencent à s'afficher, déclenche des balises de quartile (début, FirstQuartile, midpoint, ThirdQuartile, terminé) au moment opportun, charge et exécute une vérification des publicités si nécessaire, et envoie les événements à des services de vérification tiers. JavaScript viewability/verification
5. **Sondage continu** : le joueur continue d'interroger l'URL de suivi tout au long de la session afin de recevoir des métadonnées mises à jour pour les prochaines pauses publicitaires et le contenu dynamique.

Ce flux de travail active des fonctionnalités avancées telles que le compte à rebours des publicités, la fonctionnalité de clic, les publicités associées, les publicités pouvant être ignorées et l'affichage d'icônes VAST pour le respect de la confidentialité.

Rubriques

- [Activer le suivi côté client](#)
- [Paramètres du serveur publicitaire](#)
- [Paramètres de requête d'interaction avec l'origine](#)
- [Session-configured features](#)
- [Meilleures pratiques pour le suivi côté client](#)
- [Client-side schéma et propriétés de suivi des publicités](#)
- [Ad-tracking calendrier des activités](#)

- [Commandes et fonctionnalités du lecteur pour le suivi des publicités côté client](#)
- [Client-side balisage](#)
- [Mode hybride avec balises publicitaires côté serveur](#)
- [Client-side intégrations de suivi des publicités](#)
- [Pagination via des balises publicitaires avec GetTracking](#)

Activer le suivi côté client

Vous activez le suivi côté client pour chaque session. Le lecteur envoie un HTTP POST au point de terminaison du MediaTailor préfixe d'initialisation de session de la configuration. Le lecteur peut éventuellement envoyer des métadonnées supplémentaires MediaTailor à utiliser pour passer des appels publicitaires, appeler l'origine d'un manifeste et invoquer ou désactiver des MediaTailor fonctionnalités au niveau de la session.

L'exemple suivant montre la structure des métadonnées JSON :

```
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2",
    "origin_access_token": "abc123",
    "overlayAvails": "on"
  },
  "origin_access_token": "abc123",
  "overlayAvails": "on"
}
```

'adsParams' is case sensitive
key is not case sensitive
Values can contain spaces. For example, 'value 2' is an allowed value.
this is an example of a query parameter designated for the origin
'overlayAvails' is case sensitive. This is an example of a feature that is enabled at the session level.

Utilisez la MediaTailor console ou l'API pour configurer l'URL du modèle de demande ADS afin de référencer ces paramètres. Dans l'exemple suivant, quels `player_params.param1` sont les paramètres du lecteur pour `param1`, et quels `player_params.param2` sont les paramètres du lecteur pour `param2`.

```
https://my.ads.com/path?param1=[player_params.param1]&param2=[player_params.param2]
```

Paramètres du serveur publicitaire

Au niveau supérieur de la structure JSON se trouve un objet `adsParams` JSON. À l'intérieur de cet objet `key/value` se trouvent des paires que MediaTailor peuvent lire et envoyer au serveur publicitaire

dans toutes les demandes de session. MediaTailor prend en charge les serveurs publicitaires suivants :

- Gestionnaire de publicités Google
- SpringServe
- FreeWheel
- Publica

Paramètres de requête d'interaction avec l'origine

Les key/value paires réservées situées au niveau supérieur de la structure JSON, telles que, et `adsParams` `availSuppressionoverlayAvails`, ne sont pas ajoutées à l'URL de la demande d'origine sous la forme de paramètres de requête. Chaque demande de manifeste de session MediaTailor adressée à l'origine contient ces paramètres de requête. L'origine ignore les paramètres de requête externes. Par exemple, MediaTailor vous pouvez utiliser les key/value paires pour envoyer des jetons d'accès à l'origine.

Session-configured features

Utilisez la structure JSON d'initialisation de session pour activer, désactiver ou remplacer des MediaTailor fonctionnalités telles que `overlayAvails`, et. `availSuppression` `adSignaling`. Toutes les configurations de fonctionnalités transmises lors de l'initialisation de la session remplacent le paramètre au niveau de la MediaTailor configuration.

Note

Les métadonnées soumises MediaTailor lors de l'initialisation de la session sont immuables et aucune métadonnée supplémentaire ne peut être ajoutée pendant la durée de la session. Utilisez SCTE-35 des marqueurs pour enregistrer les données qui changent au cours de la session. Pour de plus amples informations, veuillez consulter [MediaTailor variables de session pour les requêtes ADS](#).

Exemple: Réalisation du suivi des publicités côté client pour HLS

```
POST mediatailorURL/v1/session/hashed-account-id/origin-id/asset-id.m3u8
```



```
{
  "adsParams": {
    "deviceType": "ipad"    # This value does not change during the session.
    "uid": "abdgfdyei-2283004-ueu"
  }
}
```

Exemple: Suivi des publicités côté client pour DASH

POST *mediatailorURL*/v1/session/*hashed-account-id*/*origin-id*/*asset-id*.mpd

```
{
  "adsParams": {
    "deviceType": "androidmobile",
    "uid": "xjhddli-9189901-uic"
  }
}
```

Paramètre du mode de rapport

Vous pouvez spécifier le mode de création de rapports lors de l'initialisation d'une session en incluant le `reportingMode` paramètre dans le corps de la demande. Ce paramètre contrôle si le suivi MediaTailor des publicités est effectué côté client ou côté serveur pour la session.

- **client**- Le joueur effectue le suivi des publicités et envoie des balises au serveur publicitaire. Il s'agit du mode par défaut si aucun `reportingMode` n'est spécifié.
- **server**- MediaTailor effectue le suivi des publicités côté serveur et envoie des balises directement au serveur publicitaire.

Exemple Initialisation de session avec mode de création de rapports côté serveur

POST *mediatailorURL*/v1/session/*hashed-account-id*/*origin-id*/*asset-id*.m3u8

```
{
  "adsParams": {
    "deviceType": "ipad",
    "uid": "abdgfdyei-2283004-ueu"
  },
  "reportingMode": "server"
}
```

Exemple Initialisation de session avec mode de reporting côté client (explicite)

POST *mediatailorURL*/v1/session/*hashed-account-id/origin-id/asset-id*.mpd

```
{
  "adsParams": {
    "deviceType": "androidmobile",
    "uid": "xjhddli-9189901-uic"
  },
  "reportingMode": "client"
}
```

Note

Le `reportingMode` paramètre est défini lors de l'initialisation de la session et ne peut pas être modifié pendant la session. Si aucun `reportingMode` est spécifié, les rapports côté client sont utilisés MediaTailor par défaut pour maintenir la compatibilité descendante.

Une réponse réussie est un HTTP 200 avec un corps de réponse. Le corps contient un objet JSON avec une `manifestUrl` et une `trackingUrl` touche. Les valeurs sont des URL relatives que le lecteur peut utiliser à la fois à des fins de lecture et de suivi des événements publicitaires.

```
{
  "manifestUrl": "/v1/dashmaster/hashed-account-id/origin-id/asset-id.m3u8?aws.sessionId=session-id",
  "trackingUrl": "/v1/tracking/hashed-account-id/origin-id/session-id"
}
```

Pour plus d'informations sur le schéma de suivi côté client, consultez. [Client-side schéma et propriétés de suivi des publicités](#)

Meilleures pratiques pour le suivi côté client

Cette section décrit les meilleures pratiques en matière de suivi côté client pour les flux de travail en MediaTailor direct et en VOD.

Workflows en direct

Interrogez le terminal de suivi à un intervalle correspondant à chaque durée cible pour HLS, ou à la période de mise à jour minimale pour DASH, afin de toujours disposer des métadonnées de suivi des

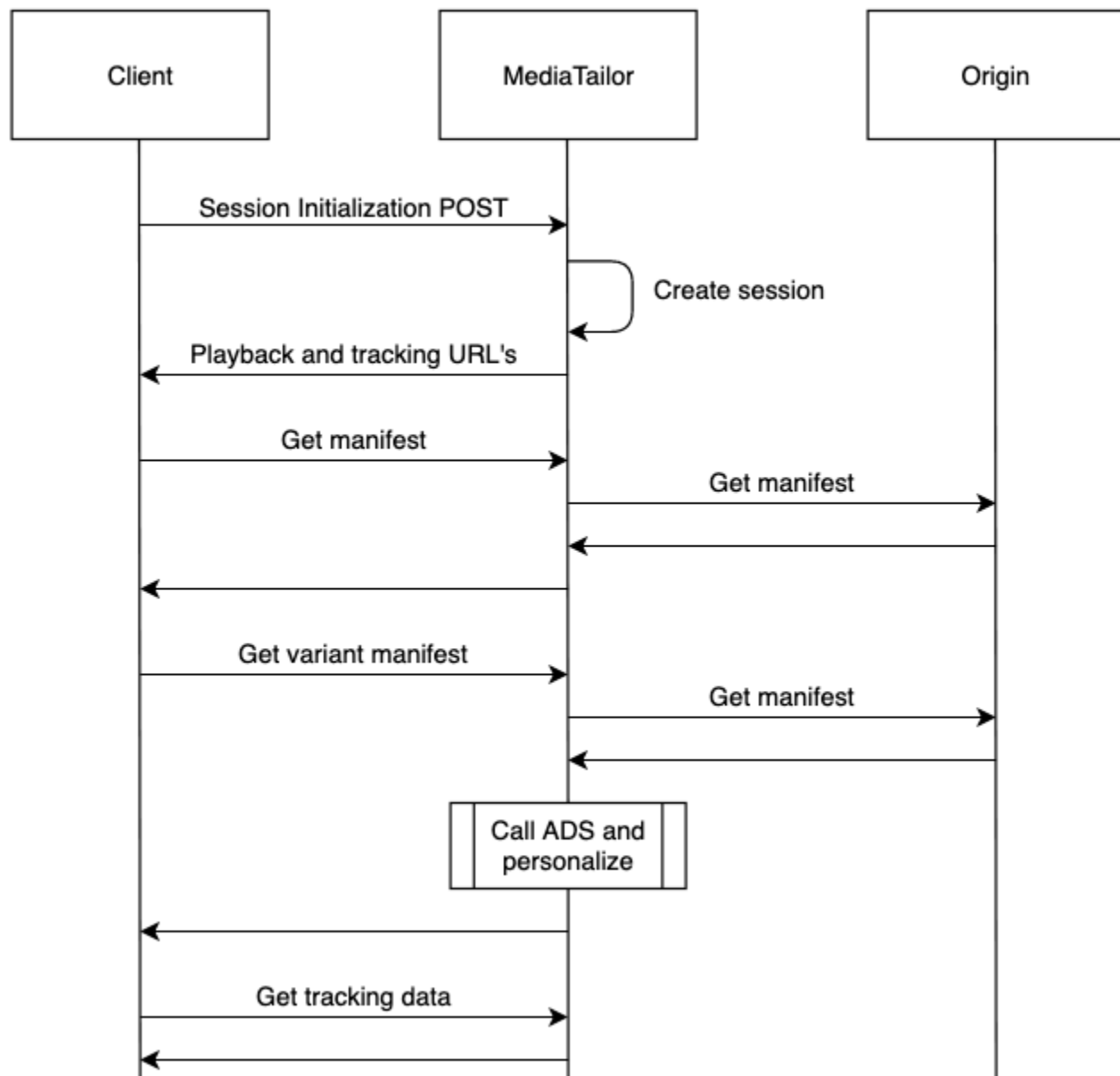
publicités les plus récentes. Il est particulièrement important de respecter cet intervalle dans les flux de travail où les créations peuvent comporter un composant interactif ou superposé.

 Note

Certains joueurs soutiennent les auditeurs des événements, ce qui pourrait être utilisé comme alternative aux sondages. Par exemple, la fonctionnalité de décoration de l'identifiant MediaTailor publicitaire doit être activée pour chaque session. Pour de plus amples informations, veuillez consulter [Décoration d'identification publicitaire](#). L'utilisation de cette fonctionnalité permet de placer un identifiant de plage de dates (HLS) ou d'élément d'événement (DASH) sur chaque annonce disponible. Les joueurs peuvent utiliser ces balises de manifeste pour les inviter à appeler le terminal MediaTailor de suivi de la session.

Flux de travail VOD

Une fois l'initialisation de la session réussie et la MediaTailor réception du premier manifeste contenant du contenu multimédia, vous ne devez appeler le terminal de suivi qu'une seule fois.



Server-guided insertion d'annonces

Server-guided les sessions d'insertion d'annonces (SGAI) n'utilisent pas l'GetTrackingAPI. Au lieu de cela, lorsque vous l'utilisez `aws.reportingMode=CLIENT`, MediaTailor fournit des informations de suivi dans la TRACKING section de chaque réponse à la liste d'actifs lorsque les joueurs demandent du contenu publicitaire. La réponse d'initialisation de session n'inclut pas `detrackingUrl`.

La réponse de la liste des actifs pour les sessions SGAI suivies côté client a la structure suivante :

```
{
  "ASSETS": [
    { "DURATION": 20.0, "URI": "https://cdn.example.com/ad1/master.m3u8" },
    { "DURATION": 10.0, "URI": "https://cdn.example.com/ad2/master.m3u8" }
  ],
  "TRACKING": {
    ...VAST tracking events and beacon URLs for each ad...
  }
}
```

Lors de la mise en œuvre du suivi côté client pour les méthodes SGAI :

- Analysez la TRACKING section à partir des réponses de la liste d'actifs plutôt que d'appeler `GetTracking`
- Utilisez les URL de suivi fournies dans la liste des actifs pour les rapports sur les événements publicitaires
- Balises de suivi des incendies basées sur les événements réels de diffusion des publicités dans le lecteur
- Gérez le suivi de chaque diffusion publicitaire indépendamment au fur et à mesure que les listes d'actifs sont récupérées

Important

La TRACKING section n'est incluse dans la liste des actifs que lorsqu'elle `aws.reportingMode=CLIENT` est définie. Lorsque les rapports côté serveur sont utilisés (valeur par défaut pour SGAI), TRACKING cette section est MediaTailor omise et les données des balises sont intégrées dans les URI des annonces à la place. Pour en savoir plus, consultez [Server-side suivi avec insertion d'annonces guidée par serveur \(SGAI\)](#).

Client-side schéma et propriétés de suivi des publicités

Grâce à la fonction de suivi des publicités MediaTailor côté client, vous pouvez intégrer des données détaillées de suivi des publicités côté client dans votre environnement de joueur. Les sections suivantes présentent le schéma général de suivi des publicités, ainsi que les propriétés et valeurs spécifiques qui le composent.

Schema

La structure JSON suivante montre le schéma de suivi des publicités MediaTailor côté client. Cette représentation illustre la structure d'imbrication du schéma pour vous aider à comprendre les relations entre les différentes parties.

Pour obtenir des informations détaillées sur chaque propriété, consultez [the section called "Propriétés"](#).

```
{
  "avails": [
    {
      "ads": [
        {
          "adID": "string",
          "adParameters": "string",
          "adSystem": "string",
          "adTitle": "string",
          "adVerifications": [
            {
              "executableResource": [
                {
                  "apiFramework": "string",
                  "type": "string",
                  "uri": "string",
                  "language": "string"
                }
              ],
              "javaScriptResource": [
                {
                  "apiFramework": "string",
                  "browserOptional": "string",
                  "uri": "string"
                }
              ],
              "trackingEvents": [
                {
                  "event": "string",
                  "uri": "string"
                }
              ],
              "vendor": "string",
              "verificationParameters": "string"
            }
          ]
        }
      ]
    }
  ]
}
```

```
    }
  ],
  "companionAds": [
    {
      "adParameters": "string",
      "altText": "string",
      "attributes": {
        "adSlotId": "string",
        "apiFramework": "string",
        "assetHeight": "string",
        "assetWidth": "string",
        "expandedHeight": "string",
        "expandedWidth": "string",
        "height": "string",
        "id": "string",
        "pxratio": "string",
        "renderingMode": "string",
        "width": "string"
      },
      "companionClickThrough": "string",
      "companionClickTracking": "string",
      "htmlResource": "string",
      "iFrameResource": "string",
      "sequence": "string",
      "staticResource": "string",
      "trackingEvents": [
        {
          "event": "string",
          "uri": "string"
        }
      ]
    }
  ],
  "creativeId": "string",
  "creativeSequence": "string",
  "duration": "string",
  "durationInSeconds": number,
  "extensions": [
    {
      "type": "string",
      "content": "string"
    }
  ],
  "icons": [
```

```

{
  "attributes": {
    "apiFramework": "string",
    "duration": "string",
    "height": "string",
    "offset": "string",
    "program": "string",
    "pxratio": "string",
    "width": "string",
    "xPosition": "string",
    "yPosition": "string"
  },
  "htmlResource": "string",
  "iconClicks": {
    "iconClickThrough": "string",
    "iconClickTracking": {
      "id": "string"
    },
    "iconClickFallbackImages": [
      {
        "altText": "string",
        "height": "string",
        "width": "string",
        "staticResource": {
          "creativeType": "string",
          "uri": "string"
        }
      }
    ]
  },
  "iconViewTracking": "string",
  "iFrameResource": "string",
  "staticResource": {
    "creativeType": "string",
    "uri": "string"
  }
},
"mediaFiles": {
  "adParameters": "string",
  "duration": "string",
  "durationInSeconds": number,
  "mediaFilesList": [
    {

```



```

        "apiFramework": "string",
        "delivery": "string",
        "height": "string",
        "maintainAspectRatio": "string",
        "mediaFileUri": "string",
        "mediaType": "string",
        "scalable": "string",
        "width": "string",
        "bitrate": "string"
    }
],
    "mezzanine": "string",
    "startTime": "string",
    "startTimeInSeconds": number,
    "trackingEvents": [
        {
            "beaconUrls": ["string"],
            "duration": "string",
            "durationInSeconds": number,
            "dateTime": "string",
            "eventId": "string",
            "eventType": "string",
            "startTime": "string",
            "startTimeInSeconds": number
        }
    ],
    "startTime": "string",
    "startTimeInSeconds": number,
    "dateTime": "string",
    "adBreakTrackingEvents": [...],
    "vastAdId": "string"
}
],
    "adType": "string",
    "availID": "string",
    "duration": "string",
    "durationInSeconds": number,
    "startTime": "string",
    "startTimeInSeconds": number,
    "dateTime": "string",
    "adMarkerDuration": "string",
    "adProgramDateTime": "string",
    "dashAvailabilityStartTime": "string",

```

```

    "hlsAnchorMediaSequenceNumber": "string"
  },
],
"nonLinearAvails": [
  {
    "nonLinearAds": [...],
    "nonLinearAdsList": [...]
  }
],
"nextToken": "string",
"meta": {}
}

```

Propriétés

Le tableau suivant répertorie les propriétés de l'API de suivi côté client, leurs définitions, leurs types de valeurs et des exemples.

Propriété	Définition	Type de la valeur	Exemple
adID	- HLS - le numéro de séquence associé au début de l'annonce - DASH - l'identifiant de période de l'annonce Parcours : <code>/avails/ads/adId</code> Cartographie VAST : Aucune	Chaîne	10
adBreakTrackingEvents	Un tableau qui contient les événements de suivi VMAP issus de la réponse VAST. Pour plus d'informations, consultez la section 2.3.3 de la spécification VMAP 1.0. Parcours : <code>/avails/ads/adBreakTrackingEvents</code>	Tableau	[]

Propriété	Définition	Type de la valeur	Exemple
<code>adMarkerDuration</code>	Durée de disponibilité observée à partir du marqueur publicitaire figurant dans le manifeste.	Chaîne	30
<code>adParameters</code>	Chaîne de paramètres publicitaires, provenant du VAST VPAID, qui est MediaTailor transmise au lecteur. Parcours : <code>/avails/ads/adParameters</code> Cartographie VAST : <code>VAST/Ad/Inline/Creatives/Creative/Linear/AdParameters</code>	Chaîne	
<code>adProgramDateTime</code>	- HLS : date, au format ISO/IEC 8601:2004, qui représente la première séquence multimédia de l'annonce. - DASH - [Non spécifié]	Chaîne	
<code>ads</code>	Tableau contenant les objets publicitaires qui constituent l'offre. Les annonces sont répertoriées dans l'ordre dans lequel elles apparaissent dans le manifeste. Parcours : <code>/avails/ads</code>	Tableau	[]

Propriété	Définition	Type de la valeur	Exemple
adSystem	Le nom du système qui diffuse l'annonce.  Important Assurez-vous de fournir une valeur. Si vous ne fournissez pas de valeur, des problèmes peuvent survenir.	Chaîne	myADS
adTitle	Titre de l'annonce.	Chaîne	ad1
adVerifications	Contient les ressources et les métadonnées nécessaires pour exécuter un code de mesure tiers afin de vérifier la lecture créative. Pour plus d'informations sur cette propriété, consultez la section 3.16 de la spécification https://iabtechlab.com/standards/vast/ VAST 4.2. MediaTailor prend adVerifications en charge les nœuds d'extension VAST 3. Parcours : /avails/ads/adVerifications Cartographie VAST : VAST/Ad/Inline/AdVerifications	Tableau	[]

Propriété	Définition	Type de la valeur	Exemple
<code>altText</code>	Texte alternatif pour l'image d'une annonce complémentaire. Ce texte permet aux joueurs disposant d'un support audio descriptif pour les malvoyants de relire une description de l'image. Parcours : <code>/avails/ads/companionAds/altText</code>	Chaîne	video sequence advertising sneakers
<code>apiFramework</code>	Régalez sur VPAID pour indiquer au joueur que cette annonce est une annonce VPAID. Peut apparaître à plusieurs endroits dans le schéma.	Chaîne	VPAID
<code>availID</code>	- HLS - le numéro de séquence associé au début de l'annonce disponible. - DASH : l'identifiant de période de disponibilité de l'annonce, qui est généralement l'identifiant de période du contenu à remplacer par une annonce. Parcours : <code>/avails/availID</code>	Chaîne	34 - PT34S_1

Propriété	Définition	Type de la valeur	Exemple
<code>avails</code>	Un tableau contenant des objets de rupture de publicité, ou avantages, qui sont présentés dans la fenêtre de manifeste active. Les ressources sont répertoriées dans l'ordre dans lequel elles apparaissent dans le manifeste. Parcours : <code>/avails</code>	Tableau	<code>[]</code>
<code>adType</code>	Le type de l'annonce. Parcours : <code>/avails/adType</code> et <code>/avails/ads/adType</code>	Chaîne	
<code>dateTime</code>	Programmez la date et l'heure, au format ISO 8601 secondes, pour le début de l'annonce ou de l'annonce. Parcours : <code>/avails/dateTime</code> et <code>/avails/ads/dateTime</code>	Chaîne	
<code>duration</code>	Durée, au format ISO 8601 secondes. La réponse inclut les durées pour l'ensemble de la diffusion de l'annonce et pour chaque annonce et balise, bien que les durées des balises soient toujours nulles. Parcours : <code>/avails/duration</code> et <code>/avails/ads/duration</code>	Chaîne	<code>15.015</code>

Propriété	Définition	Type de la valeur	Exemple
<code>durationInSeconds</code>	Durée, au format secondes. Parcours : <code>/avails/durationInSeconds</code> et <code>/avails/ads/durationInSeconds</code>	Nombre	
<code>extensions</code>	Extensions personnalisées de VAST utilisées par les serveurs publicitaires. Pour plus d'informations sur les extensions, consultez la section 3.18 de la spécification https://iabtechlab.com/standards/vast/ VAST 4.2. Parcours : <code>/avails/ads/extensions</code> Cartographie VAST : <code>VAST/Ad/Inline/Extensions</code>	Tableau	<code>[]</code>
<code>icons</code>	Éléments d'icône pour l'annonce. Parcours : <code>/avails/ads/icons</code> Cartographie VAST : <code>VAST/Ad/Inline/Creatives/Creative/Linear/Icons</code>	Tableau	
<code>mediaFiles</code>	Les vidéos et autres ressources dont le joueur a besoin pour diffuser la publicité. Parcours : <code>/avails/ads/mediaFiles</code>	Objet	
<code>nonLinearAvails</code>	Tableau d'objets non linéaires et disponibles. Parcours : <code>/nonLinearAvails</code>	Tableau	

Propriété	Définition	Type de la valeur	Exemple
<code>executableResource</code>	Ressources exécutables pour la vérification. Parcours : <code>/avails/ads/adVerifications/executableResource</code> Cartographie VAST : <code>VAST/Ad/InLine/AdVerifications/Verification/ExecutableResource</code>	Tableau	
<code>javascriptResource</code>	JavaScript ressources pour la vérification. Parcours : <code>/avails/ads/adVerifications/javascriptResource</code> Cartographie VAST : <code>VAST/Ad/InLine/AdVerifications/Verification/JavaScriptResource</code>	Tableau	
<code>trackingEvents</code>	Suivi des événements à des fins de vérification ou d'annonces complémentaires. Trajectoire : <code>/avails/ads/adVerifications/trackingEvents</code> ou <code>/avails/ads/companionAds/trackingEvents</code>	Tableau	

Propriété	Définition	Type de la valeur	Exemple
vendor	Fournisseur de services de vérification. Parcours : /avails/ads/adVerifications/vendor Cartographie VAST : VAST/Ad/Inline/AdVerifications/Verification/@vendor	Chaîne	
uri	URI qui pointe vers une ressource exécutable, une ressource vidéo ou un point de terminaison de suivi. Chemin : différents emplacements dans le schéma Cartographie VAST : divers éléments CDATA dans VAST	Chaîne	https://tracking.example.com/impression
verificationParameters	Paramètres de vérification Parcours : /avails/ads/adVerifications/verificationParameters Cartographie VAST : VAST/Ad/Inline/AdVerifications/Verification/VerificationParameters	Chaîne	
attributes	Attributs d'annonces complémentaires tels que les dimensions et le mode de rendu. Parcours : /avails/ads/companionAds/attributes	Objet	

Propriété	Définition	Type de la valeur	Exemple
<code>companionClickThrough</code>	URL de la page de l'annonceur que le lecteur multimédia ouvre lorsque le spectateur clique sur l'annonce associée. Parcours : <code>/avails/ads/companionAds/companionClickThrough</code> Cartographie VAST : <code>VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickThrough</code>	Chaîne	<code>https://aws.amazon.com/</code>
<code>companionClickTracking</code>	URL de suivi de la <code>companionClickThrough</code> propriété. Parcours : <code>/avails/ads/companionAds/companionClickTracking</code> Cartographie VAST : <code>VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickTracking</code>	Chaîne	<code>https://myads.com/beaconing/event=clicktracking</code>

Propriété	Définition	Type de la valeur	Exemple
htmlResource	Le CDATA-encoded code HTML inséré directement dans la page HTML du fournisseur de streaming. Parcours : /avails/ads/companionAds/htmlResource Cartographie VAST : VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/HTMLResource	Chaîne	<pre><![CDATA[<!doctype html><htm l><head>< meta name= \"viewport \" content= \"width=1, initial-s cale=1.0, minimum-s cale=1.0, ...]]></pre>
iFrameResource	URL d'un fichier de ressources HTML que le fournisseur de streaming charge dans un iframe. Parcours : /avails/ads/companionAds/iFrameResource Cartographie VAST : VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/iFrameResource	Chaîne	
sequence	La valeur de séquence spécifiée pour la création dans la réponse VAST. Parcours : /avails/ads/companionAds/sequence	Chaîne	1

Propriété	Définition	Type de la valeur	Exemple
<code>startTime</code>	La position temporelle, au format ISO 8601 secondes. Pour HLS, cela se rapporte au début de la session de lecture. Pour DASH, cela est relatif à l'AST (heure de début de disponibilité) du manifeste. La réponse inclut les heures de début pour l'ensemble de la diffusion de publicités, et pour chaque publicité et balise beacon. Parcours : <code>/avails/startTime</code> et <code>/avails/ads/startTime</code>	Chaîne	PT18.581355S
<code>startTimeInSeconds</code>	Position temporelle, au format secondes. Pour HLS, cela se rapporte au début de la session de lecture. Pour DASH, cela est relatif à l'AST (heure de début de disponibilité) du manifeste. La réponse inclut les heures de début pour l'ensemble de la diffusion de publicités, et pour chaque publicité et balise beacon. Parcours : <code>/avails/startTimeInSeconds</code> et <code>/avails/ads/startTimeInSeconds</code>	Nombre	18.581
<code>eventId</code>	- HLS : numéro de séquence associé à la balise. - DASH : <code>ptsTime</code> date du début de l'annonce.	Chaîne	

Propriété	Définition	Type de la valeur	Exemple
event	Le nom de l'événement de suivi. Trajectoire : /avails/ads/adVerifications/trackingEvents/event ou /avails/ads/companionAds/trackingEvents/event	Chaîne	impression , start, firstQuartile , midpoint, thirdQuartile , complete
beaconUrls	URL à laquelle MediaTailor est envoyée la balise publicitaire. Parcours : /avails/ads/mediaFiles/trackingEvents/beaconUrls	Tableau	
bitrate	Le débit de la ressource vidéo. Cette propriété n'est généralement pas incluse pour une ressource exécutable.	Chaîne	2048
companionAds	Une ou plusieurs spécifications de contenu publicitaire complémentaire, chacune d'entre elles spécifiant un fichier de ressources à utiliser. Des publicités complémentaires accompagnent l'annonce et fournissent du contenu, tel qu'un cadre autour de l'annonce ou une bannière, à afficher à proximité de la vidéo. Parcours : /avails/ads/companionAds	Tableau	[]
creativeId	La valeur d'Idattribut de la Creative balise de l'annonce.	Chaîne	creative-1

Propriété	Définition	Type de la valeur	Exemple
<code>creativeSequence</code>	L'ordre dans lequel une annonce doit être diffusée, en fonction de la <code>AdId</code> valeur de la réponse VAST.	Chaîne	1
<code>dashAvailabilityStartTime</code>	Pour live/dynamic DASH, le <code>MPD@availabilityStartTime</code> manifeste d'origine.	Chaîne	2022-10-05T19:38:39.263Z
<code>delivery</code>	Indique si un streaming protocole progressive ou est utilisé.	Chaîne	progressive
<code>eventType</code>	Type de balise. Parcours : <code>/avails/ads/mediaFiles/trackingEvents/eventType</code>	Chaîne	impression
<code>height</code>	Hauteur, en pixels, de la ressource vidéo.	Chaîne	360
<code>hlsAnchorMediaSequenceNumber</code>	Le numéro de séquence multimédia de la séquence first/oldest multimédia vue dans le manifeste d'origine HLS.	Chaîne	77
<code>maintainAspectRatio</code>	Indique s'il faut conserver le format d'image de la vidéo lors de la mise à l'échelle.	Booléen	true
<code>mediaFilesList</code>	Spécifie la vidéo et les autres ressources dont le joueur a besoin pour diffuser la publicité. Parcours : <code>/avails/ads/mediaFiles/mediaFilesList</code>	Tableau	[]

Propriété	Définition	Type de la valeur	Exemple
<code>mediaFileUri</code>	URI qui pointe vers une ressource exécutable ou une ressource vidéo.	Chaîne	<code>https://myad.com/ad/ad134/vpaid.js</code>
<code>mediaType</code>	Type MIME de la ressource créative ou complémentaire.	Chaîne	<code>video/mp4</code>
<code>meta</code>	Métadonnées supplémentaires pour l'annonce.	Objet	
<code>mezzanine</code>	URL de la ressource MP4 de la mezzanine, spécifiée si l'annonce VPAID en inclut une. Parcours : <code>/avails/ads/mediaFiles/mezzanine</code>	Chaîne	<code>https://gcdn.2mdn.net/video/playback/id/itag/c/k2/file/file.mp4</code>
<code>nextToken</code>	La valeur du jeton qui pointe vers la page de résultats suivante, lorsqu'une telle valeur existe.	Chaîne	<code>UFQz0S44NzNTXzIwMjMtMDctMzFUMTY6NTA6MDYuMzUwNjI2ODQ1Wl8x</code>
<code>nonLinearAds</code>	Non-linear publicités qui apparaissent à côté du contenu vidéo.	Tableau	<code>[]</code>
<code>nonLinearAdsList</code>	Liste des publicités non linéaires.	Tableau	<code>[]</code>
<code>scalable</code>	Indique s'il faut redimensionner la vidéo en fonction d'autres dimensions.	Booléen	<code>true</code>

Propriété	Définition	Type de la valeur	Exemple
<code>skipOffset</code>	La valeur temporelle qui identifie le moment où le joueur met les commandes de saut à la disposition de l'utilisateur.	Chaîne	<code>00:00:05</code>
<code>staticResource</code>	URL d'un fichier créatif statique utilisé pour le composant publicitaire. Parcours : <code>/avails/ads/companionAds/staticResource</code>	Chaîne	<code>https://very-interactive-ads.com/campaign1/file.json?c=1019113602</code>
<code>vastAdId</code>	La valeur d'Idattribut de la Ad balise.	Chaîne	<code>ad1</code>
<code>width</code>	Largeur, en pixels, de la ressource vidéo.	Chaîne	<code>640</code>
<code>xPosition</code>	Position horizontale d'une icône dans le lecteur vidéo. Il peut s'agir d'une valeur de pixel spécifique ou d'une position telle que « gauche » ou « droite ». Parcours : <code>/avails/ads/icons/attributes/xPosition</code>	Chaîne	<code>left</code> ou <code>10</code>
<code>yPosition</code>	Position verticale d'une icône dans le lecteur vidéo. Il peut s'agir d'une valeur de pixel spécifique ou d'une position telle que « haut » ou « bas ». Parcours : <code>/avails/ads/icons/attributes/yPosition</code>	Chaîne	<code>top</code> ou <code>10</code>

Propriété	Définition	Type de la valeur	Exemple
<code>iconClicks</code>	Contient les informations relatives aux clics et au suivi d'une icône. Parcours : <code>/avails/ads/icons/iconClicks</code>	Objet	
<code>iconClickThrough</code>	URL de la page de l'annonceur que le lecteur multimédia ouvre lorsque le spectateur clique sur l'icône. Parcours : <code>/avails/ads/icons/iconClicks/iconClickThrough</code>	Chaîne	<code>https://advertiser.com/landing-page</code>
<code>iconClickTracking</code>	URL de suivi de la <code>iconClickThrough</code> propriété. Parcours : <code>/avails/ads/icons/iconClicks/iconClickTracking</code>	Objet	
<code>iconClickFallbackImages</code>	Tableau d'images de repli à afficher si l'icône ne peut pas être affichée. Parcours : <code>/avails/ads/icons/iconClicks/iconClickFallbackImages</code>	Tableau	
<code>iconViewTracking</code>	URL permettant de suivre l'affichage d'une icône. Parcours : <code>/avails/ads/icons/iconViewTracking</code>	Chaîne	<code>https://tracking.example.com/icon-view</code>

Propriété	Définition	Type de la valeur	Exemple
offset	Décalage temporel pendant lequel une icône doit apparaître pendant la diffusion d'une annonce. Parcours : /avails/ads/icons/attributes/offset	Chaîne	00:00:05
program	Le programme ou l'initiative associé à l'icône, tel que « AdChoices ». Parcours : /avails/ads/icons/attributes/program	Chaîne	AdChoices
pxratio	Rapport de pixels de l'icône ou de la publicité associée, utilisé pour les écrans haute résolution. Chemin : /avails/ads/icons/attributes/pxratio ou /avails/ads/companionAds/attributes/pxratio	Chaîne	1 ou 2
type	Type de ressource ou d'extension. Chemin : /avails/ads/extensions/type ou /avails/ads/adVerifications/executableResource/type	Chaîne	text/javascript
content	Le contenu d'une extension. Parcours : /avails/ads/extensions/content	Chaîne	

Propriété	Définition	Type de la valeur	Exemple
language	Langage de programmation d'une ressource exécutable. Parcours : /avails/ads/adVerifications/executableResource/language	Chaîne	javascript
browserOptional	Indique si la prise en charge du navigateur est requise pour la JavaScript ressource. Parcours : /avails/ads/adVerifications/javascriptResource/browserOptional	Chaîne	true ou false
id	Un identifiant pour les différents éléments du schéma. Chemin : /avails/ads/companionAds/attributes/id ou /avails/ads/icons/iconClicks/iconClickTracking/id	Chaîne	companion-1
assetHeight	Hauteur de l'élément publicitaire associé. Parcours : /avails/ads/companionAds/attributes/assetHeight	Chaîne	250

Propriété	Définition	Type de la valeur	Exemple
<code>assetWidth</code>	Largeur de la ressource publicitaire associée. Parcours : <code>/avails/ads/companionAds/attributes/assetWidth</code>	Chaîne	<code>300</code>
<code>expandedHeight</code>	Hauteur de l'annonce d'accompagnement lorsqu'elle est agrandie. Parcours : <code>/avails/ads/companionAds/attributes/expandedHeight</code>	Chaîne	<code>600</code>
<code>expandedWidth</code>	Largeur de l'annonce complémentaire lorsqu'elle est agrandie. Parcours : <code>/avails/ads/companionAds/attributes/expandedWidth</code>	Chaîne	<code>600</code>
<code>renderingMode</code>	Mode de rendu de l'annonce complémentaire. Parcours : <code>/avails/ads/companionAds/attributes/renderingMode</code>	Chaîne	<code>default</code> ou <code>transparent</code>
<code>adSlotId</code>	L'identifiant de l'espace publicitaire dans lequel l'annonce complémentaire doit être affichée. Parcours : <code>/avails/ads/companionAds/attributes/adSlotId</code>	Chaîne	<code>banner-1</code>

Propriété	Définition	Type de la valeur	Exemple
<code>creativeType</code>	Type MIME de la ressource créative. Parcours : <code>/avails/ads/icons/staticResource/creativeType</code>	Chaîne	<code>image/png</code>

Ad-tracking calendrier des activités

Avec les rapports côté client, le joueur doit émettre des événements de suivi (balises) avec un certain niveau de précision. À l'aide du schéma de suivi MediaTailor côté client, vous pouvez vous assurer que, pour chaque événement d'annonce, d'accompagnement, de superposition et de suivi, les informations relatives au calendrier et à la durée sont présentes, sous différentes formes.

Utilisez les MediaTailor key/value paires suivantes pour que le lecteur réconcilie avec précision les activités liées aux événements publicitaires, telles que le suivi des événements, avec la position de lecture :

- [`startTime`](#)
- [`startTimeInSeconds`](#)
- [`adProgramDateTime`](#)
- [`adId/eventId`](#)

HLS et DASH mettent en œuvre la valeur de `startTime` et `startTimeInSeconds` différemment :

- HLS - Les `startTime` valeurs sont relatives au début de la session de lecture. Le début de la session de lecture est défini comme l'heure zéro. L'annonce `startTime` est la somme des valeurs cumulées de toutes les durées des EXT-INF segments qui ont précédé la mise en vente. Le numéro de séquence multimédia du segment auquel se rapporte l'annonce ou l'événement de suivi correspond également à la réponse `adId` de suivi côté client. `eventId`
- DASH :
 - Live/dynamic manifestes - Les `startTime` valeurs sont relatives à celles `MPD@availabilityStartTime` du manifeste DASH. `MPD@availabilityStartTime` s'agit d'un point d'ancrage temporel pour toutes les MediaTailor sessions qui consomment le flux.

- VOD/static manifestes : les `startTime` valeurs sont relatives au début de la session de lecture. Le début de la session de lecture est défini comme l'heure zéro. Chaque annonce contenue dans le produit est contenue dans son propre `Period` élément. L'`Period` élément possède un `@start` attribut dont la valeur est identique à celle de la `startTime` charge utile de suivi côté client. Cela correspond `PeriodId` également au `adId` ou `eventId` dans la réponse de suivi côté client.

Exemple HLS :

Dans l'exemple suivant, la MediaTailor session a démarré et le manifeste suivant est le premier transmis au client :

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:4603263
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:23.295678Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_34.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:27.306345Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_35.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:31.317012Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_36.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:35.327679Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_37.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:39.338346Z
#EXTINF:2.538667,
https://123.cloudfront.net/out/v1/index_1_38.ts
#EXT-X-DISCONTINUITY
#EXT-X-KEY:METHOD=NONE
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:41.453Z
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00001.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:43.453Z
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00002.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:45.453Z
```

```
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00003.ts
```

Dans la charge utile JSON de suivi côté client, les valeurs suivantes s'appliquent :

- `startTime`: "PT18.581355S"
- `startTimeInSeconds`: 18.581
- `availProgramDateTime`: "2023-05-03T21:24:41.453Z"
- `adId`: 4603269

Exemple DASH :

Dans l'exemple suivant, la MediaTailor session obtient un midroll dans le manifeste. Notez que la valeur d'`@startattribut` de la deuxième période, qui est la période de l'annonce, possède une valeur relative à la `MPD@availabilityStartTime` valeur. Cette valeur est celle qui est écrite dans MediaTailor les `startTime` champs de réponse de suivi côté client, pour toutes les sessions.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD availabilityStartTime="2022-10-05T19:38:39.263Z" minBufferTime="PT10S"
  minimumUpdatePeriod="PT2S" profiles="urn:mpeg:dash:profile:isoff-live:2011"
  publishTime="2023-05-03T22:06:48.411Z" suggestedPresentationDelay="PT10S"
  timeShiftBufferDepth="PT1M30S" type="dynamic" xmlns="urn:mpeg:dash:schema:mpd:2011"
  xmlns:scte35="urn:scte:scte35:2013:xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://123.channel-assembly.mediataylor.us-west-2.amazonaws.com/v1/
  channel/my-channel/</BaseURL>
  <Location>https://123.cloudfront.net/v1/
  dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/MediaTailor-Live-HLS-DASH/channel/
  channel1/dash.mpd?aws.sessionId=794a15e0-2a7f-4941-a537-9d71627984e5</Location>
  <Period id="1683151479166_1" start="PT5042H25M59.903S"
  xmlns="urn:mpeg:dash:schema:mpd:2011">
    <BaseURL>https://123.cloudfront.net/out/v1/f1a946be8efa45b0931ea35c9055fb74/
    ddb73bf548a44551a0059c346226445a/eea5485198bf497284559efb8172425e/</BaseURL>
    <AdaptationSet ...>
      ...
    </AdaptationSet>
  </Period>
  <Period id="1683151599194_1_1" start="PT5042H27M59.931S">
    <BaseURL>https://123.cloudfront.net/
    tm/94063eadf7d8c56e9e2edd84fdf897826a70d0df/fpc5omz5wzd2rdepgieibp23ybyqyrme/</BaseURL>
```

```
<AdaptationSet ...>
  ...
</AdaptationSet>
</Period>
</MPD>
```

Dans la charge utile JSON de suivi côté client, les valeurs suivantes s'appliquent :

- `startTime`: "PT5042H27M59.931S"
- `startTimeInSeconds`: 18152879.931
- `availProgramDateTime`: *null*
- `adId`: 1683151599194_1_1

Commandes et fonctionnalités du lecteur pour le suivi des publicités côté client

MediaTailor les métadonnées de suivi côté client prennent en charge diverses commandes et fonctionnalités du lecteur. La liste suivante décrit les commandes de lecteur les plus courantes.

Rubriques

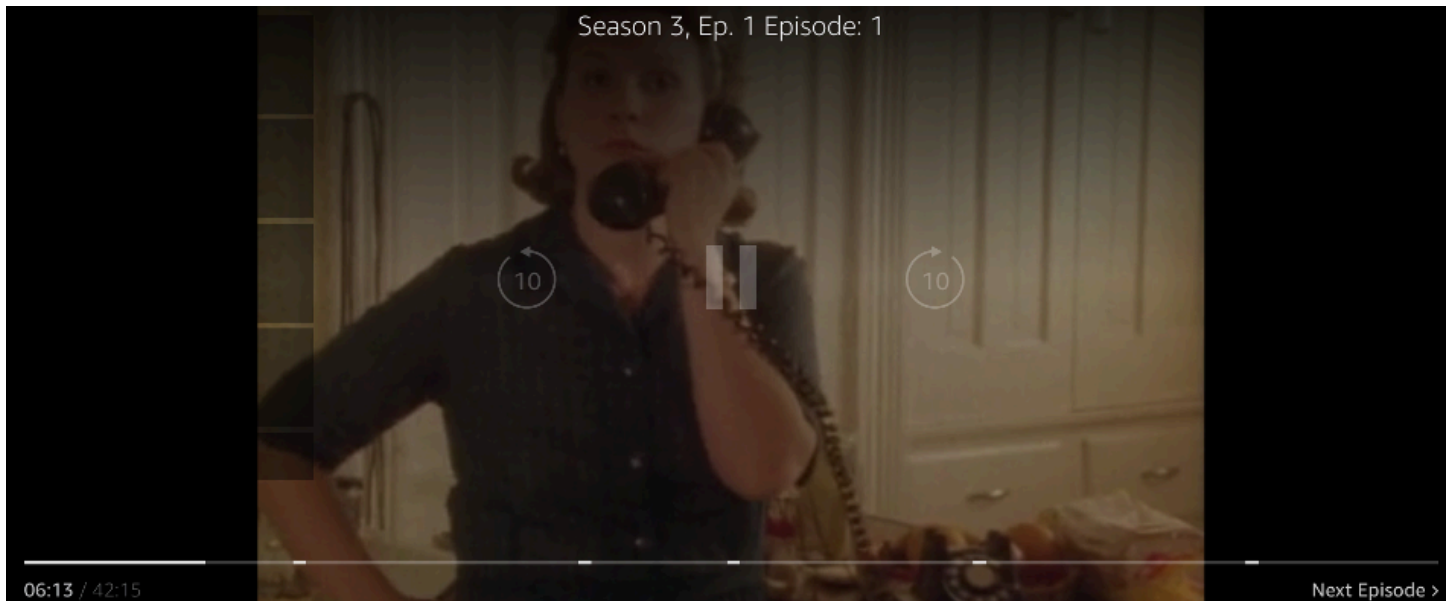
- [Gommage](#)
- [Compte à rebours publicitaire](#)
- [Annonces pouvant être ignorées](#)
- [Cliquez sur une annonce](#)
- [Annonces complémentaires](#)
- [Publicités interactives \(SIMID\)](#)
- [Publicités interactives \(VPAID\)](#)
- [Icônes pour Google Why This Ad \(WTA\)](#)

Gommage

Pour améliorer l'expérience de lecture, le lecteur peut afficher les positions des annonces dans la chronologie de lecture. MediaTailor rend ces positions d'annonces disponibles sous forme de `startTimeInSeconds` valeurs dans la réponse de suivi côté client.

Note

Certains fournisseurs de streaming empêchent de passer la position d'une annonce à l'autre.



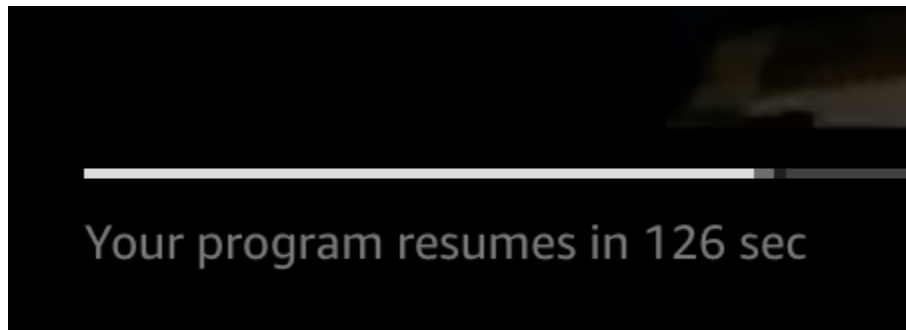
La réponse JSON de charge utile de suivi côté client suivante indique l'heure de début de l'avail (ad break) dans l'objet JSON racine du tableau avails. Le joueur utilise ces données pour afficher l'emplacement de la pause publicitaire sur la chronologie du joueur, à 28 secondes.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [...],
      "availId": "7",
      "availProgramDateTime": null,
      "duration": "PT30S",
      "durationInSeconds": 30,
      "meta": null,
      "nonLinearAdsList": [],
      "startTime": "PT28S",
      "startTimeInSeconds": 28
    }
  ],
  "dashAvailabilityStartTime": null,
```

```
"hlsAnchorMediaSequenceNumber": null,  
"nextToken": "UFQxMk0zNC44NjhTXzIwMjMtMDctMjFUMjA6MjM6MDcuNzc1NzE2MzAyWl8x",  
"nonLinearAvails": []  
}
```

Compte à rebours publicitaire

Avec, MediaTailor vous pouvez utiliser un compte à rebours pour maintenir l'engagement de votre public pendant le visionnage des pauses publicitaires. Le public peut utiliser le chronomètre pour comprendre quand la pause publicitaire prend fin et quand son programme reprend.



Les éléments des métadonnées de suivi côté client qui jouent un rôle dans le compte à rebours des publicités sont `startTime`, `startTimeInSeconds`, `duration`, et `durationInSeconds`. Le lecteur utilise ces métadonnées, ainsi que la durée de la session qu'il suit séparément, pour déterminer quand afficher le chronomètre et la valeur à partir de laquelle le compte à rebours doit être effectué.

La réponse JSON de charge utile de suivi côté client suivante indique les informations nécessaires à l'affichage d'un compte à rebours publicitaire.

```
{  
  "avails": [  
    {  
      "adBreakTrackingEvents": [],  
      "adMarkerDuration": null,  
      "ads": [...],  
      "availId": "7",  
      "availProgramDateTime": null,  
      "duration": "PT30S",  
      "durationInSeconds": 30,  
      "meta": null,  
      "nonLinearAdsList": [],  
      "startTime": "PT28S",  
      "startTimeInSeconds": 28  
    }  
  ]  
}
```

```

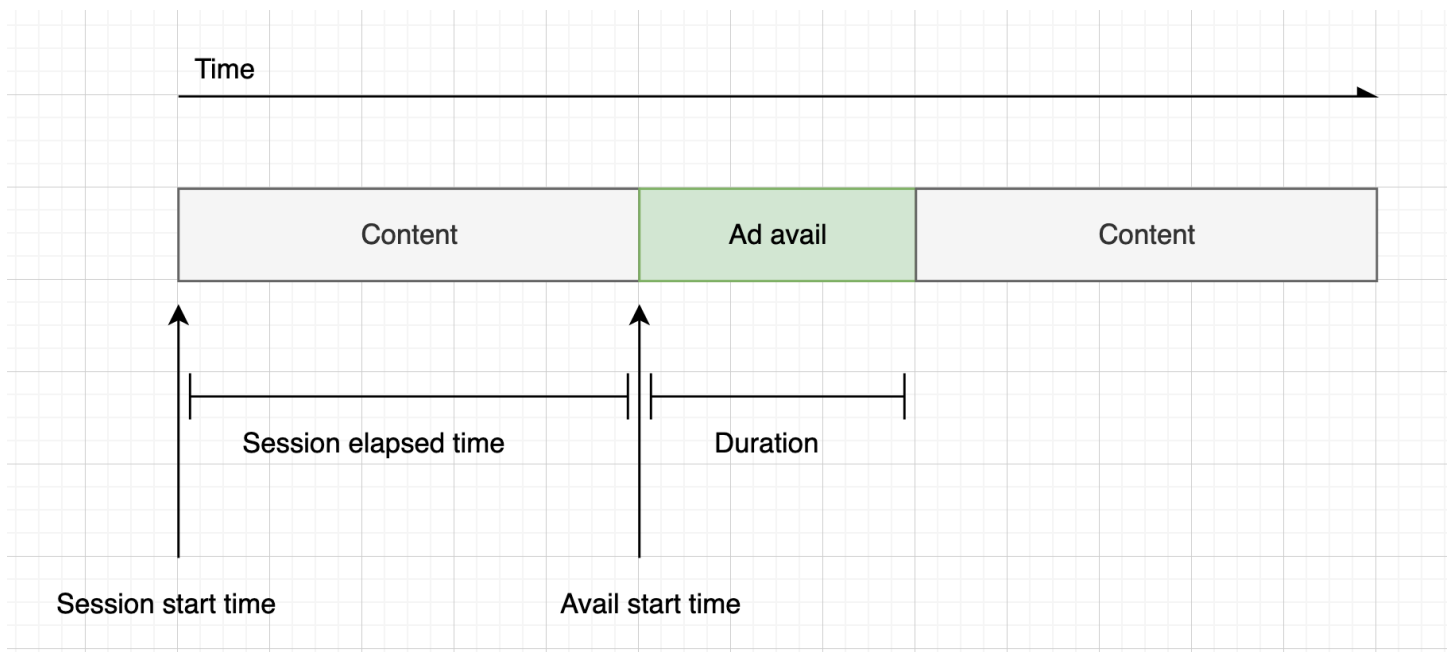
    }
  ],
  "dashAvailabilityStartTime": null,
  "hlsAnchorMediaSequenceNumber": null,
  "nextToken": "UFQxMk0zNC44NjhTXzIwMjMtMDctMjFUMjA6MjM6MDcuNzc1NzE2MzAyWl8x",
  "nonLinearAvails": []
}

```

Lorsque la durée de la session atteint l'heure de début de la session, le joueur affiche un compte à rebours dont la valeur correspond à la durée de la session. La valeur du compte à rebours diminue au fur et à mesure que le temps écoulé dépasse l'heure de début de la disponibilité.

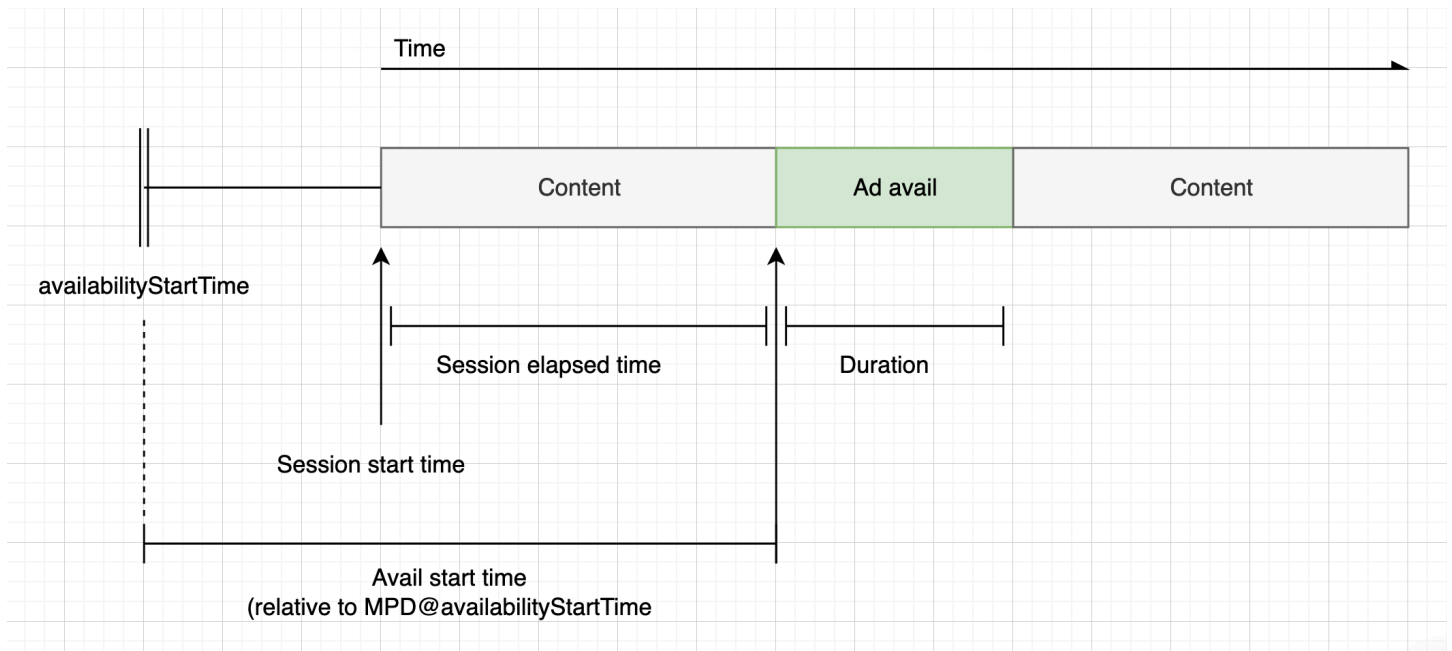
Exemple formule : compte à rebours pour HLS (live et VOD) et DASH (VOD)

- $\text{session\_start\_time} = \text{la somme de toutes les valeurs de EXT-INF durée} - \text{la valeur de durée des trois séquences EXT-INF multimédia les plus récentes}$
- $\text{valeur du minuteur} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Exemple formule : compte à rebours pour DASH (en direct)

- $\text{session\_start\_time} = (\text{startTime} + \text{du segment le plus récent} \times \text{duration}) / \text{timescale} - \text{MPD@suggestedPresentationDelay}$
- $\text{valeur du minuteur} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Annonces pouvant être ignorées

Les publicités ignorables sont des spots publicitaires qui permettent au spectateur d'ignorer une partie de l'annonce pour reprendre le visionnage du programme. Dans VAST, l'attribut `Linear@skipOffset` identifie une annonce pouvant être ignorée.

La réponse VAST suivante montre comment utiliser une annonce pouvant être ignorée :

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <Inline>
      ...
      <Creatives>
        <Creative id="1" sequence="1">
          <Linear skipoffset="00:00:05">
            <Duration>00:00:15</Duration>
            <MediaFiles>
              <MediaFile id="EMT" delivery="progressive" width="640" height="360"
                type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![
                CDATA[https://ads.com/file.mp4]]></MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
```

```

    ...
  </InLine>
</Ad>
</VAST>

```

La réponse JSON de charge utile de suivi côté client suivante affiche les métadonnées de l'annonce dans le tableau. ads Le tableau contient la skipOffset valeur MediaTailor obtenue à partir de la réponse VAST.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:11:40.693Z",
          "adSystem": "2.0",
          "adTitle": "AD-skiing-15",
          "adVerifications": [],
          "companionAds": [...],
          "creativeId": "1",
          "creativeSequence": "1",
          "duration": "PT15.015S",
          "durationInSeconds": 15.015,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "skipOffset": "00:00:05",
          "startTime": "PT9.943S",
          "startTimeInSeconds": 9.943,
          "trackingEvents": [
            {
              "beaconUrls": [
                "https://adserverbeaconing.com/v1/impression"
              ],
              "duration": "PT15.015S",
              "durationInSeconds": 15.015,
              "eventId": "2697726",

```

```

        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT9.943S",
        "startTimeInSeconds": 9.943
      }
    ],
    "vastAdId": ""
  }
],
"availId": "2697726",
"availProgramDateTime": "2023-07-31T16:11:40.693Z",
"duration": "PT15.015S",
"durationInSeconds": 15.015,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT9.943S",
"startTimeInSeconds": 9.943
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "",
"nonLinearAvails": []
}

```

Cliquez sur une annonce

Click-through Les URI permettent aux annonceurs de mesurer la capacité d'une annonce à capter l'attention des spectateurs. Lorsqu'un internaute clique sur l'image vidéo active d'une annonce en cours, un navigateur Web ouvre l'URI de la page d'accueil ou de la page de destination de la campagne de l'annonceur. Le développeur du lecteur détermine le comportement des clics, par exemple en superposant un bouton ou une étiquette sur la vidéo publicitaire, avec un message sur lequel cliquer pour en savoir plus. Les développeurs de lecteurs interrompent souvent la vidéo de l'annonce une fois que les spectateurs ont cliqué sur l'image vidéo active.



Click here for deals on Amazon.com

MediaTailor peut analyser et mettre à disposition toutes les URL d'événements de clic vidéo linéaires renvoyées dans la réponse VAST. La réponse VAST suivante montre un exemple de clic sur une annonce.

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <Inline>
      ...
    <Creatives>
      <Creative id="1" sequence="1">
        <Linear>
          <Duration>00:00:15</Duration>
          <MediaFiles>
```

```

        <MediaFile id="EMT" delivery="progressive" width="1280" height="720"
        type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![
[CDATA[https://ads.com/file.mp4]]></MediaFile>
        </MediaFiles>
        <VideoClicks>
            <ClickThrough id="EMT"><![CDATA[https://aws.amazon.com]]></ClickThrough>
            <ClickTracking id="EMT"><![CDATA[https://myads.com/beaconing/
event=clicktracking]]></ClickTracking>
        </VideoClicks>
    </Linear>
</Creative>
</Creatives>
...
</InLine>
</Ad>
</VAST>

```

La réponse JSON de charge utile de suivi côté client suivante montre comment MediaTailor s'affiche les URL de clic et de suivi des clics dans le tableau. `trackingEvents` Le type `clickThrough` d'événement représente l'annonce cliquable et le type `clickTracking` événement représente l'URL de suivi des clics.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "1",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "00006",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],

```



```
    "mezzanine": ""
  },
  "skipOffset": null,
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339,
  "trackingEvents": [
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=impression"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "impression",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://aws.amazon.com"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "clickThrough",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=clicktracking"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698795",
      "eventProgramDateTime": null,
      "eventType": "clickTracking",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    }
  ],
  "vastAdId": ""
```

```

    }
  ],
  "availId": "2698188",
  "availProgramDateTime": "2023-07-31T16:53:40.577Z",
  "duration": "PT14.982S",
  "durationInSeconds": 14.982,
  "meta": null,
  "nonLinearAdsList": [],
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}

```

Annonces complémentaires

Une publicité complémentaire apparaît à côté d'une création linéaire. Utilisez des publicités complémentaires pour améliorer l'efficacité d'un spot publicitaire en affichant des informations sur le produit, le logo et la marque. L'annonce graphique peut comporter des codes de réponse rapide (QR) et des zones cliquables pour favoriser l'engagement du public.

MediaTailor prend en charge les publicités complémentaires dans la réponse VAST. Il peut transmettre des métadonnées depuis `StaticResource` et `FrameResource`, et des `HTMLResource` nœuds, respectivement.

La réponse VAST suivante montre un exemple d'emplacement et de format de l'annonce linéaire et de l'annonce complémentaire.

```

<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <Inline>
      ...
    <Creatives>
      <Creative id="1" sequence="1">
        <Linear>
          <Duration>00:00:10</Duration>

```

```

    <MediaFiles>
      <MediaFile id="EMT" delivery="progressive" width="640" height="360"
        type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![
[CDATA[https://ads.com/file.mp4]]></MediaFile>
    </MediaFiles>
  </Linear>
</Creative>
<Creative id="2" sequence="1">
  <CompanionAds>
    <Companion id="2" width="300" height="250">
      <StaticResource creativeType="image/png"><![CDATA[https://emt.com/
companion/9973499273]]></StaticResource>
      <TrackingEvents>
        <Tracking event="creativeView"><![CDATA[https://beacon.com/1]]></
Tracking>
      </TrackingEvents>
      <CompanionClickThrough><![CDATA[https://beacon.com/2]]></
CompanionClickThrough>
    </Companion>
    <Companion id="3" width="728" height="90">
      <StaticResource creativeType="image/png"><![CDATA[https://emt.com/
companion/1238901823]]></StaticResource>
      <TrackingEvents>
        <Tracking event="creativeView"><![CDATA[https://beacon.com/3]]></
Tracking>
      </TrackingEvents>
      <CompanionClickThrough><![CDATA[https://beacon.com/4]]></
CompanionClickThrough>
    </Companion>
  </CompanionAds>
</Creative>
</Creatives>
...
</InLine>
</Ad>
</VAST>

```

Les données apparaissent dans la réponse de suivi côté client dans la `/avail/x/ads/y/` `companionAds` liste. Chaque création linéaire peut contenir jusqu'à 6 publicités complémentaires. Comme le montre l'exemple ci-dessous, les publicités associées apparaissent dans une liste

Note

Il est recommandé aux développeurs d'applications de mettre en œuvre une logique permettant de supprimer ou de télécharger explicitement l'annonce associée à la fin de la création.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "EMT",
          "adTitle": "sample",
          "adVerifications": [],
          "companionAds": [
            {
              "adParameters": null,
              "altText": null,
              "attributes": {
                "adSlotId": null,
                "apiFramework": null,
                "assetHeight": null,
                "assetWidth": null,
                "expandedHeight": null,
                "expandedWidth": null,
                "height": "250",
                "id": "2",
                "pxratio": null,
                "renderingMode": null,
                "width": "300"
              },
              "companionClickThrough": "https://beacon.com/2",
              "companionClickTracking": null,
              "htmlResource": null,
              "iFrameResource": null,
              "sequence": "1",
```

```
    "staticResource": "https://emt.com/companion/9973499273",
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/1"
        ],
        "eventType": "creativeView"
      }
    ]
  },
  {
    "adParameters": null,
    "altText": null,
    "attributes": {
      "adSlotId": null,
      "apiFramework": null,
      "assetHeight": null,
      "assetWidth": null,
      "expandedHeight": null,
      "expandedWidth": null,
      "height": "90",
      "id": "3",
      "pxratio": null,
      "renderingMode": null,
      "width": "728"
    },
    "companionClickThrough": "https://beacon.com/4",
    "companionClickTracking": null,
    "htmlResource": null,
    "iFrameResource": null,
    "sequence": "1",
    "staticResource": "https://emt.com/companion/1238901823",
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/3"
        ],
        "eventType": "creativeView"
      }
    ]
  }
],
"creativeId": "1",
"creativeSequence": "1",
```

```

    "duration": "PT10S",
    "durationInSeconds": 10,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": null,
    "startTime": "PT0S",
    "startTimeInSeconds": 0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/impression/1"
        ],
        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": ""
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNFmJyAyMy0wNy0wNlQyMT0xMDowOC42NzQ4NDA1NjJaXzE%3D",
"nonLinearAvails": []
}

```

Publicités interactives (SIMID)

SecureInteractive La définition de l'interface multimédia (SIMID) est une norme pour la publicité interactive qui a été introduite dans la norme VAST 4.x de l'Interactive Advertising Bureau (IAB). SIMID dissocie le chargement des éléments interactifs de la création linéaire principale sur le lecteur, en faisant référence à ces deux éléments dans la réponse VAST. MediaTailor intègre la création principale pour préserver l'expérience de lecture et place les métadonnées des composants interactifs dans la réponse de suivi côté client.

Dans l'exemple de réponse VAST 4 suivant, la charge utile SIMID se trouve à l'intérieur du InteractiveCreativeFile nœud.

```
<?xml version="1.0"?>
<VAST xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <Inline>
      <AdSystem>SampleAdSystem</AdSystem>
      <AdTitle>Linear SIMID Example</AdTitle>
      <Description>SIMID example</Description>
      <Error>https://www.beacons.com/error</Error>
      <Impression>https://www.beacons.com/impression</Impression>
      <Creatives>
        <Creative sequence="1">
          <Linear>
            <Duration>00:00:15</Duration>
            <TrackingEvents>
              ...
            </TrackingEvents>
            <VideoClicks>
              <ClickThrough id="123">https://aws.amazon.com</ClickThrough>
              <ClickTracking id="123">https://www.beacons.com/click</ClickTracking>
            </VideoClicks>
            <MediaFiles>
              <MediaFile delivery="progressive" type="video/mp4">
https://interactive-ads.com/interactive-media-ad-sample/media/file.mp4
              </MediaFile>
              <InteractiveCreativeFile type="text/html" apiFramework="SIMID"
                variableDuration="true">
https://interactive-ads.com/interactive-media-ad-sample/sample\_simid.html
              </InteractiveCreativeFile>
```

```

        </MediaFiles>
    </Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

Dans la réponse VAST 3 suivante, la charge utile SIMID se trouve à l'intérieur du Extensions nœud.

```

<?xml version="1.0"?>
<VAST xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <InLine>
      <AdSystem>SampleAdSystem</AdSystem>
      <AdTitle>Linear SIMID Example</AdTitle>
      <Description>SIMID example</Description>
      <Impression>https://www.beacons.com/impression</Impression>
      <Creatives>
        <Creative id="1" sequence="1">
          <Linear>
            <Duration>00:00:15</Duration>
            <TrackingEvents>
              ...
            </TrackingEvents>
            <VideoClicks>
              <ClickThrough id="123">https://aws.amazon.com</ClickThrough>
              <ClickTracking id="123">https://myads.com/beaconing/event=clicktracking</
ClickTracking>
            </VideoClicks>
            <MediaFiles>
              <MediaFile delivery="progressive" type="video/mp4">
https://interactive-ads.com/interactive-media-ad-
sample/media/file.mp4
              </MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
      <Extensions>
        <Extension type="InteractiveCreativeFile">

```



```

        <InteractiveCreativeFile type="text/html" apiFramework="SIMID"
variableDuration="true">
            https://interactive-ads.com/interactive-media-ad-sample/sample_simid.html
        </InteractiveCreativeFile>
    </Extension>
</Extensions>
</InLine>
</Ad>
</VAST>

```

Dans la réponse de suivi côté client suivante, les données SIMID apparaissent dans la liste. /
 avails/x/ads/y/extensions

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "Linear SIMID Example",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "1",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [
            {
              "content": "<InteractiveCreativeFile type=\"text/html\" apiFramework=
\"SIMID\" variableDuration=\"true\">\nhttps://interactive-ads.com/interactive-media-ad-
sample/sample_simid.html</InteractiveCreativeFile>",
              "type": "InteractiveCreativeFile"
            }
          ],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          }
        }
      ]
    }
  ]
}

```

```
"skipOffset": null,
"startTime": "PT39.339S",
"startTimeInSeconds": 39.339,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=impression"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "impression",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://aws.amazon.com"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "clickThrough",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=clicktracking"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698795",
    "eventProgramDateTime": null,
    "eventType": "clickTracking",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"vastAdId": ""
}
```

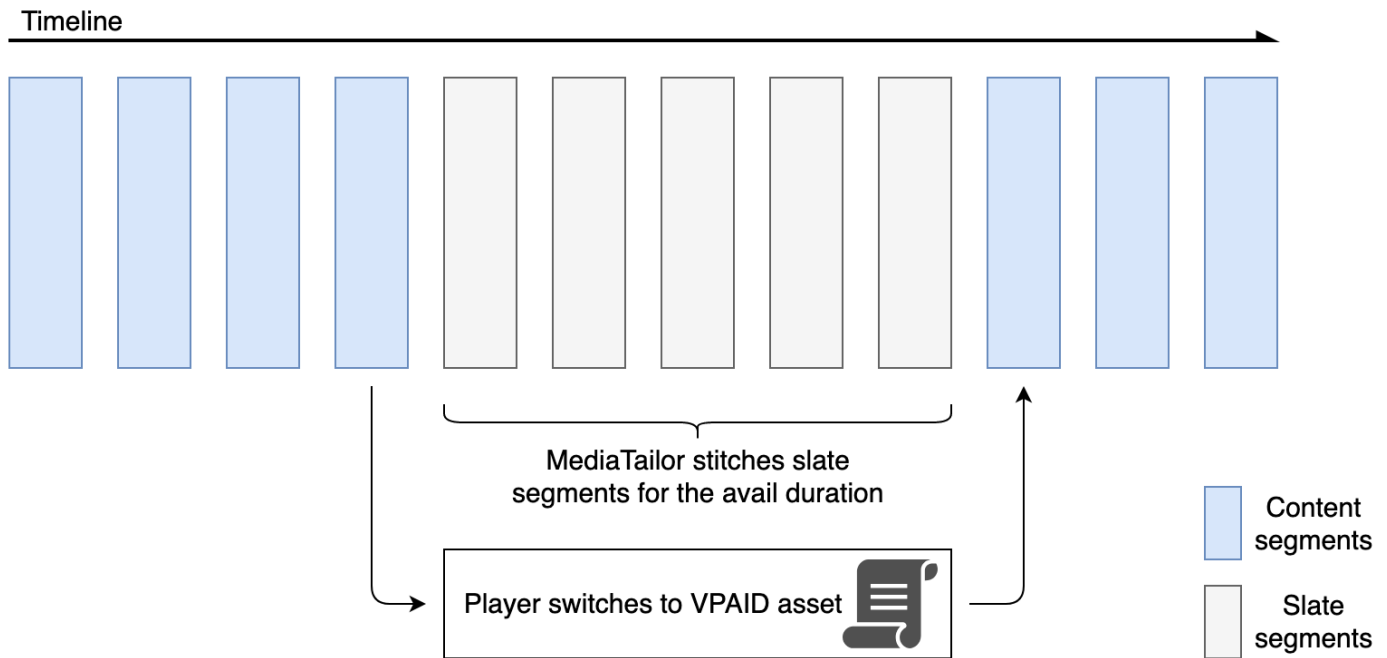
```
    "availId": "2698188",
    "availProgramDateTime": "2023-07-31T16:53:40.577Z",
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}
```

Publicités interactives (VPAID)

La définition de l'interface publicitaire du lecteur vidéo (VPAID) spécifie le protocole entre l'annonce et le lecteur vidéo qui permet l'interactivité des publicités et d'autres fonctionnalités. Pour les diffusions en direct, MediaTailor prend en charge le format VPAID en assemblant des segments d'ardoise pendant la durée de disponibilité et en insérant des métadonnées pour les créations VPAID dans la réponse de suivi côté client consommée par le lecteur vidéo. Le lecteur télécharge les fichiers VPAID, lit la création linéaire et exécute les scripts du client. Le joueur ne doit jamais jouer les segments de l'ardoise.

Note

VPAID est obsolète depuis VAST 4.1.



L'exemple suivant montre le contenu VPAID de la réponse VAST.

```
<?xml version="1.0"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <Inline>
      <AdSystem>GDFP</AdSystem>
      <AdTitle>VPAID</AdTitle>
      <Description>Vpaid Linear Video Ad</Description>
      <Error>http://www.example.com/error</Error>
      <Impression>http://www.example.com/impression</Impression>
      <Creatives>
        <Creative sequence="1">
          <Linear>
            <Duration>00:00:00</Duration>
            <TrackingEvents>
              <Tracking event="start">http://www.example.com/start</Tracking>
              <Tracking event="firstQuartile">http://www.example.com/firstQuartile</
Tracking>
              <Tracking event="midpoint">http://www.example.com/midpoint</Tracking>
              <Tracking event="thirdQuartile">http://www.example.com/thirdQuartile</
Tracking>
```

```

    <Tracking event="complete">http://www.example.com/complete</Tracking>
    <Tracking event="mute">http://www.example.com/mute</Tracking>
    <Tracking event="unmute">http://www.example.com/unmute</Tracking>
    <Tracking event="rewind">http://www.example.com/rewind</Tracking>
    <Tracking event="pause">http://www.example.com/pause</Tracking>
    <Tracking event="resume">http://www.example.com/resume</Tracking>
    <Tracking event="fullscreen">http://www.example.com/fullscreen</Tracking>
    <Tracking event="creativeView">http://www.example.com/creativeView</
Tracking>
    <Tracking event="acceptInvitation">http://www.example.com/
acceptInvitation</Tracking>
  </TrackingEvents>
  <AdParameters><![CDATA[ {"videos":[ {"url":"https://my-ads.com/interactive-
media-ads/media/media_linear_VPAID.mp4", "mimetype":"video/mp4"}]} ]]></AdParameters>
  <VideoClicks>
    <ClickThrough id="123">http://google.com</ClickThrough>
    <ClickTracking id="123">http://www.example.com/click</ClickTracking>
  </VideoClicks>
  <MediaFiles>
    <MediaFile delivery="progressive" apiFramework="VPAID" type="application/
javascript" width="640" height="480"> https://googleads.github.io/googleads-ima-html5/
vpaid/linear/VpaidVideoAd.js </MediaFile>
  </MediaFiles>
</Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

L'exemple suivant montre les informations de suivi.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",

```

```
"adTitle": "1",
"adVerifications": [],
"companionAds": [],
"creativeId": "00006",
"creativeSequence": "1",
"duration": "PT14.982S",
"durationInSeconds": 14.982,
"extensions": [],
"mediaFiles": {
  "mediaFilesList": [],
  "mezzanine": ""
},
"skipOffset": null,
"startTime": "PT39.339S",
"startTimeInSeconds": 39.339,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=impression"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "impression",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://aws.amazon.com"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "clickThrough",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=clicktracking"
    ],
```

```

        "duration": "PT14.982S",
        "durationInSeconds": 14.982,
        "eventId": "2698795",
        "eventProgramDateTime": null,
        "eventType": "clickTracking",
        "startTime": "PT39.339S",
        "startTimeInSeconds": 39.339
      }
    ],
    "vastAdId": ""
  }
],
"availId": "2698188",
"availProgramDateTime": "2023-07-31T16:53:40.577Z",
"duration": "PT14.982S",
"durationInSeconds": 14.982,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT39.339S",
"startTimeInSeconds": 39.339
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "2922274",
          "adParameters": "",
          "adProgramDateTime": "2023-08-14T19:49:53.998Z",
          "adSystem": "Innovid Ads",
          "adTitle": "VPAID",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "",
          "creativeSequence": "",
          "duration": "PT16.016S",
          "durationInSeconds": 16.016,

```

```

    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [
        {
          "apiFramework": "VPAID",
          "bitrate": 0,
          "codec": null,
          "delivery": "progressive",
          "height": 9,
          "id": "",
          "maintainAspectRatio": false,
          "maxBitrate": 0,
          "mediaFileUri": "http://my-ads.com/mobileapps/js/vpaid/1h41kg?
cb=178344c0-8e67-281a-58ca-962e4987cd60&deviceid=&ivc=",
          "mediaType": "application/javascript",
          "minBitrate": 0,
          "scalable": false,
          "width": 16
        }
      ],
      "mezzanine": "http://my-ads.com/mobileapps/js/vpaid/1h41kg?
cb=178344c0-8e67-281a-58ca-962e4987cd60&deviceid=&ivc="
    },
    "skipOffset": null,
    "startTime": "PT8M42.289S",
    "startTimeInSeconds": 522.289,
    "trackingEvents": [
      {
        "beaconUrls": [
          "about:blank"
        ],
        "duration": "PT16.016S",
        "durationInSeconds": 16.016,
        "eventId": "2922274",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT8M42.289S",
        "startTimeInSeconds": 522.289
      }
    ],
    "vastAdId": "1h41kg"
  }
],
  "availId": "2922274",

```



```

    "availProgramDateTime": "2023-08-14T19:49:53.998Z",
    "duration": "PT16.016S",
    "durationInSeconds": 16.016,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT8M42.289S",
    "startTimeInSeconds": 522.289
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQ4TTQyLjI4OVNfMjAyMy0wOC0xNFQxOT01MD00MS4zOTc5MjAzODVaXzE%3D",
"nonLinearAvails": []
}

```

Icônes pour Google Why This Ad (WTA)

AdChoices est une norme du secteur qui fournit aux internautes des informations sur les publicités qu'ils voient, notamment sur la manière dont ces publicités les ciblent.



L'API de suivi MediaTailor côté client prend en charge les métadonnées des icônes contenues dans le nœud d'extensions VAST de la réponse VAST. Pour plus d'informations sur le WTA dans la réponse VAST, consultez [cet exemple de réponse](#) XML VAST.

Note

MediaTailor ne prend actuellement en charge que la version 3 de VAST.

```

<VAST>
  <Ad>
    <Inline>
      ...
    <Extensions>
      <Extension type="IconClickFallbackImages">

```

```

    <IconClickFallbackImages program="GoogleWhyThisAd">
      <IconClickFallbackImage width="400" height="150">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://storage.googleapis.com/interactive-media-ads/images/wta_dialog.png]]></StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>
    <IconClickFallbackImages program="AdChoices">
      <IconClickFallbackImage width="400" height="150">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=1x]]></StaticResource>
      </IconClickFallbackImage>
      <IconClickFallbackImage width="800" height="300">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=2x]]></StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>
  </Extension>
</Extensions>
</InLine>
</Ad>
</VAST>

```

L'exemple suivant montre la réponse de suivi côté client dans la `/avails/x/ads/y/extensions` liste.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "GDFP",
          "adTitle": "Google Why This Ad VAST 3 Sample",
          "adVerifications": [],

```

```

"companionAds": [],
"creativeId": "7891011",
"creativeSequence": "1",
"duration": "PT10S",
"durationInSeconds": 10,
"extensions": [
  {
    "content": "<IconClickFallbackImages program=\"GoogleWhyThisAd\">
      <IconClickFallbackImage height=\"150\" width=\"400\">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png]]>
      </StaticResource>
    </IconClickFallbackImage>
    </IconClickFallbackImages>
    <IconClickFallbackImages program=\"AdChoices\">
      <IconClickFallbackImage height=\"150\" width=\"400\">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=1x]]>
      </StaticResource>
    </IconClickFallbackImage>
    <IconClickFallbackImage height=\"300\" width=\"800\">
      <AltText>Alt icon fallback</AltText>
      <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=2x]]>
    </StaticResource>
    </IconClickFallbackImage>
    </IconClickFallbackImages>",
    "type": "IconClickFallbackImages"
  }
],
"mediaFiles": {
  "mediaFilesList": [],
  "mezzanine": ""
},
"skipOffset": "00:00:03",
"startTime": "PT0S",
"startTimeInSeconds": 0,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://example.com/view"
    ]
  }
],

```

```

        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": "123456"
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNfMjAyMy0wNy0wNlQyMDo0MT0xNy45NDE4MDM0NDhaXzE%3D",
"nonLinearAvails": []
}

```

Client-side balisage

Avec l'`startTimeInSeconds` élément de suivi côté client, vous pouvez l'utiliser MediaTailor pour prendre en charge la synchronisation des balises.

La réponse JSON suivante indique les principaux types de balises : impressions, début, quartiles et fin.

Note

Les directives de mesure des impressions vidéo de l'Interactive Advertising Bureau (IAB) stipulent qu'une impression nécessite que le contenu de l'annonce soit chargé côté client et, au minimum, qu'il commence à s'afficher dans le lecteur. Pour plus d'informations, consultez le modèle de diffusion d'annonces vidéo [numériques \(VAST\)](#) sur le site Web de l'IAB.

```
{
  "avails": [
    {
      "ads": [
        {
          "adId": "8104385",
          "duration": "PT15.100000078S",
          "durationInSeconds": 15.1,
          "startTime": "PT17.817798612S",
          "startTimeInSeconds": 17.817,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=impression"
              ],
              "duration": "PT15.100000078S",
              "durationInSeconds": 15.1,
              "eventId": "8104385",
              "eventType": "impression",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=start"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
              "eventId": "8104385",
              "eventType": "start",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=firstQuartile"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
              "eventId": "8104386",
              "eventType": "firstQuartile",
              "startTime": "PT21.592798631S",
              "startTimeInSeconds": 21.592
            }
          ]
        }
      ]
    }
  ]
}
```

```
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=midpoint"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104387",
      "eventType": "midpoint",
      "startTime": "PT25.367798651S",
      "startTimeInSeconds": 25.367
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=thirdQuartile"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104388",
      "eventType": "thirdQuartile",
      "startTime": "PT29.14279867S",
      "startTimeInSeconds": 29.142
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=complete"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104390",
      "eventType": "complete",
      "startTime": "PT32.91779869S",
      "startTimeInSeconds": 32.917
    }
  ]
}
],
"availId": "8104385",
"duration": "PT15.100000078S",
"durationInSeconds": 15.1,
"startTime": "PT17.817798612S",
"startTimeInSeconds": 17.817
}
]
```

```
}
```

Mode hybride avec balises publicitaires côté serveur

MediaTailor prend en charge un mode hybride pour le suivi des sessions. Dans ce mode, le service émet des événements de suivi des publicités liés à la lecture, mais met la charge utile complète de suivi côté client à disposition pour la session

Pour activer le suivi hybride à l'aide de préfixes de lecture, depuis le lecteur, initialisez une nouvelle session de MediaTailor lecture à l'aide d'une requête dans l'un des formats suivants, conformément à votre protocole :

Exemple: format HLS

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "reportingMode": "server"
}
```

Exemple: format DASH

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "reportingMode": "server"
}
```

MediaTailor gère les événements de suivi suivants en mode hybride :

- Impression
- Démarrer
- Premier quartile
- Point médian
- Troisième quartile
- Complet

- `breakStart(vmap)`
- `breakEnd(vmap)`

Client-side intégrations de suivi des publicités

Cette section décrit les intégrations entre MediaTailor et divers serveurs de suivi des publicités côté client.

Rubriques

- [Open Measurement SDK](#)
- [SDK pour lecteurs gratuits Datazoom](#)
- [Cadre publicitaire Roku \(RAF\)](#)
- [TheoPlayer](#)
- [MediaTailor SDK](#)

Open Measurement SDK

Le SDK Open Measurement (OM SDK) de l'Interactive Advertising Bureau (IAB) facilite la mesure de la visibilité et de la vérification par des tiers pour les publicités diffusées dans des environnements de vidéos Web et d'applications natives.

Pour les anciens documents VAST version 3, le code de vérification doit être chargé avec le nœud `Extension`, avec le type d'extension `AdVerifications`. La racine du nœud d'extension est un `AdVerifications` nœud dont le schéma est le même que celui de l'élément VAST 4.1.

Pour faciliter l'adoption du SDK OM, MediaTailor a établi un partenariat avec Datazoom pour fournir des SDK gratuits pour les joueurs qui sont configurés et vérifiés pour Open Measurement. Pour de plus amples informations, veuillez consulter [SDK pour lecteurs gratuits Datazoom](#).

Note

MediaTailor ne prend actuellement en charge que la version 3 de VAST.

Exemple: nœud de vérification dans VAST 3, avant la version 4.1

```
...  
<Extensions>
```



```

<Extension type="AdVerifications">
  <AdVerifications>
    <Verification vendor="company.com-omid">
      <JavaScriptResource apiFramework="omid" browserOptional="true">
        <![CDATA[https://verification.com/omid_verification.js]]>
      </JavaScriptResource>
      <TrackingEvents>
        <Tracking event="verificationNotExecuted">
          <![CDATA[https://verification.com/trackingurl]]>
        </Tracking>
      </TrackingEvents>
      <VerificationParameters>
        <![CDATA[verification params key/value pairs]]>
      </VerificationParameters>
    </Verification>
  </AdVerifications>
</Extension>
</Extensions>

```

MediaTailor extrait les AdVerifications données du <Extensions> nœud et les place dans le adVerifications tableau dans la réponse de suivi côté client.

Exemple: tableau ADVerifications dans la réponse de suivi côté client

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "3062770",
          "adParameters": "",
          "adProgramDateTime": "2023-08-23T16:25:40.914Z",
          "adSystem": "2.0",
          "adTitle": "AD-polarbear-15",
          "adVerifications": [
            {
              "executableResource": [],
              "javaScriptResource": [
                {
                  "apiFramework": "omid",
                  "browserOptional": "true",

```

[illegible]

```

        "durationInSeconds": 14.982,
        "eventId": "3062770",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT10.11S",
        "startTimeInSeconds": 10.11
      }
    ],
    "vastAdId": ""
  }
],
"availId": "3062770",
"availProgramDateTime": "2023-08-23T16:25:40.914Z",
"duration": "PT14.982S",
"durationInSeconds": 14.982,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT10.11S",
"startTimeInSeconds": 10.11
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMC4xMVNfMjAyMy0wOC0yM1QxNjoyNjoyNC4yNDYxMDIxOTBaXzE%3D",
"nonLinearAvails": []
}

```

Note

Collaborez avec l'IAB Tech Lab pour vous assurer que les applications sont certifiées chaque année afin de garantir la conformité.

Pour plus d'informations sur le SDK OM, consultez [Open Measurement SDK](#) sur le site Web de l'IAB Tech Lab.

SDK pour lecteurs gratuits Datazoom

Pour faciliter l'adoption des SDK pour les joueurs, MediaTailor a établi un partenariat avec Datazoom pour fournir des SDK gratuits pour les joueurs qui sont configurés et testés avec et avec la [Client-side AWS Elemental MediaTailor intégration avec Google Ad Manager](#) technologie IAB. [Open Measurement SDK](#)

Le SDK du lecteur Datazoom prend en charge les fonctionnalités suivantes :

- Playlists en direct et VOD
- Spécifications DASH et HLS
- Support des fournisseurs de lecteurs pour Bitmovin, exoplayer, lecteur multimédia Android, Apple AVPlayer, Brightcove, Chromecast Receiver, hls.js, JWPlayer, lecteur Dash.js Shaka, lecteur THEO, Roku, etc. Video.js
- Certification IAB Tech Lab Open Measurement, si disponible sur certains appareils
- Click-through gestion des événements
- Ad-event les répartiteurs, tels que les comptes à rebours des publicités, les événements non linéaires et superposés d'annonces, le début de la pause publicitaire, la fin de la pause publicitaire
- Client-side et balisage
- SDK PAL (Google Programmatic Access Library), en tant que paramètre de configuration facultatif

Datazoom propose également un service d'analyse et de télémétrie payant pris en charge par les SDK des joueurs. Les clients peuvent activer et contrôler la télémétrie du SDK des joueurs depuis la console de gestion Datazoom. Pour accéder aux SDK du lecteur Datazoom et en savoir plus sur le service de télémétrie et d'analyse à valeur ajoutée, utilisez les coordonnées figurant sur le site Datazoom. <https://www.datazoom.io/partner-aws>

Cadre publicitaire Roku (RAF)

Le Roku Ad Framework (RAF) assure une expérience publicitaire cohérente sur l'ensemble de la plateforme Roku. Toutes les chaînes, y compris les publicités vidéo, doivent répondre aux exigences de certification de Roku pour la RAF. Notamment, l'application doit toujours utiliser le déclenchement d'événements côté client via la RAF. MediaTailor, en tant que fournisseur d'insertion d'annonces côté serveur (SSAI), prend en charge le déclenchement d'événements côté client. Les adaptateurs RAFX SSAI fournissent des interfaces à la fois avec les serveurs de manifestes SSAI, ou Stitchers, et avec RAF. Ces interfaces incluent :

- Analyse de la masterURL réponse et extraction playURL, AdURL, et des métadonnées de l'annonce.
- Transformation des métadonnées publicitaires MediaTailor SSAI en métadonnées RAF-usable publicitaires et configuration de la RAF pour la lecture.
- Observation des événements des flux et des métadonnées chronométrées.

- Faire correspondre les événements de diffusion, les métadonnées des publicités et les pixels de l'événement de déclenchement à temps.
- Pinging/polling leAdURL, comme l'exige le serveur de manifeste MediaTailor SSAI, puis l'analyse et la reconfiguration de RAF.

Pour plus d'informations sur les adaptateurs SSAI pour RAF, consultez la section [Mise en œuvre de l'insertion d' Server-Side annonces à l'aide d'adaptateurs Roku](#) sur le site Web de Roku.

TheoPlayer

TheoPlayer l'intégration avec MediaTailor effectue les opérations suivantes :

- Fournit des fonctionnalités permettant de prendre en charge le suivi des événements MediaTailor côté client pour HLS et DASH pour les flux de travail en direct et de VOD.
- Prend en charge l'envoi de balises de suivi uniquement pour les publicités linéaires.
- Désactive la recherche pendant une annonce. Cependant, aucune logique n'est en place pour diffuser une annonce lorsque l'utilisateur cherche après la pause publicitaire.

Pour plus d'informations sur SSAI in TheoPlayer et pour passer en revue les SDK Web, Android, iOS et tvOS pour MediaTailor, consultez [MediaTailor](#) le site Web. TheoPlayer

MediaTailor SDK

AWS Elemental gère un kit de JavaScript-based développement logiciel (SDK). AWS Elemental fournit le SDK tel quel, sans garantie implicite. Utilisez le SDK comme démonstration de référence pour rationaliser votre intégration à l'utilisation. MediaTailor Le SDK montre comment interagir avec l'API de suivi MediaTailor côté client. Le SDK met en œuvre le suivi des publicités côté client et la création de rapports pour les joueurs. HTML5-based Le SDK initialise une session de reporting MediaTailor côté client, puis demande périodiquement des informations de suivi des publicités. Pendant la lecture, le SDK émet des événements de suivi des publicités lorsque de nouveaux événements publicitaires sont détectés.

Le MediaTailor SDK prend en charge les fonctionnalités suivantes :

- Playlists en direct et VOD
- Spécifications DASH et HLS
- Click-through gestion des événements

- Ad-event répartiteurs
- Crochets événementiels personnalisés
- Client-side et balisage. Pour plus d'informations sur l'envoi de balises publicitaires, consultez [Client-side balisage](#).

 Note

Soumettez un ticket d' AWS assistance pour recevoir un exemple de JavaScript SDK pour MediaTailor. Vous recevrez un lien de téléchargement pour le package et ses fichiers.

Pagination via des balises publicitaires avec GetTracking

Utilisez le `GetTracking` point de terminaison pour réduire le nombre d'annonces renvoyées à un joueur. Par exemple, si la fenêtre d'un manifeste est large et s'étend sur une longue période, le nombre de balises publicitaires renvoyées peut avoir un impact sur les performances des joueurs.

`GetTracking` renvoie une `NextToken` valeur que vous pouvez utiliser pour réduire le nombre de balises renvoyées en parcourant la liste des balises renvoyées. Vous pouvez parcourir les `NextToken` valeurs pour trouver la valeur souhaitée dans le `StartTimeInSeconds` champ d'une balise publicitaire.

- Lors du premier appel à `GetTracking`, toutes les publicités possibles figurant dans la fenêtre du manifeste sont renvoyées, y compris un `NextToken` et une valeur pour chacune.
- Si une `GetTracking` demande n'inclut pas de `NextToken`, toutes les annonces de la fenêtre du manifeste sont renvoyées.
- Si une `GetTracking` demande contient un `NextToken` mais qu'il n'y a aucune nouvelle balise à renvoyer, MediaTailor renvoie la même valeur que celle `NextToken` que vous avez envoyée lors de la demande d'origine.
- Lorsqu'il n'y a plus de beacons correspondant à une annonce, `GetTracking` supprime l'annonce de sa réponse.
- Les jetons `GetTracking` expirent au bout de 24 heures. Si une `NextToken` valeur date de plus de 24 heures, le prochain appel à `GetTracking` renvoie une valeur nulle. `NextToken`

Séquence d'appel généralisée GetTracking du joueur sortant

Émanant du joueur client, une GetTracking requête est un POST dont le corps de requête contient les publicités NextToken et les balises associées au jeton.

```
https://YouMediaTailorUrl/v1/tracking
{
    "NextToken": "value"
    .
    .
    .
}
```

La séquence générale d'utilisation GetTracking avec NextToken est la suivante :

1. Passez le premier appel àGetTracking.

Toutes les publicités et balises, ainsi que les premières NextToken pour les appels suivants, sont renvoyées.

2. Si la valeur de NextToken est nulle, MediaTailor renvoie toutes les balises publicitaires.
3. Si le NextToken est expiré, MediaTailor renvoie un message d'erreur HTTP avec le code de retour 400.

Passez un nouvel appel à GetTracking pour récupérer des NextToken s valides.

4. Scannez l'intégralité de la réponse pour trouver celle StartTimeInSeconds d'une balise publicitaire se trouvant dans la plage souhaitée.
5. Effectuez un nouvel appel à GetTracking avec la valeur de NextToken associée à la valeur souhaitéeStartTimeInSeconds.
6. Si nécessaire, parcourez à nouveau les publicités renvoyées jusqu'à ce que vous trouviez exactement celles auxquelles vous souhaitez jouer.

Exemple étendu

Cet exemple montre comment utiliser S GetTracking NextToken pour limiter le nombre de balises publicitaires renvoyées à un joueur.

MediaTailor reçoit une GetTracking demande. La réponse contient une annonce avec l'ID 9935407 et deux balises avec des StartTimeInSeconds valeurs de 52,286 et 48,332 secondes.

MediaTailor envoie la réponse JSON NextToken comme suit :

```
{
  "NextToken": JF57ITe48t1441mv7TmLKuZLroxDzfIslp6BiSNL1IJmzPVMDN0lqrBYycgMbKEb
  "avails": [
    {
      "ads": [
        {
          "adId": "9935407",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "",
          "creativeSequence": "",
          "duration": "PT15S",
          "durationInSeconds": 15,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "startTime": "PT30S",
          "StartTimeInSeconds": 45,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://adserver.com/tracking?event=Impression "
              ],
              "duration": "PT0S",
              "durationInSeconds": 0,
              "eventId": "9935414",
              "eventType": "secondQuartile",
              "startTime": "PT52.286S",
              "StartTimeInSeconds": 52.286
            },
            {
              "beaconUrls": [
                "http://adserver.com/tracking?event=firstQuartile"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0,
```



```

        "eventId": "9935412",
        "eventType": "firstQuartile",
        "startTime": "PT48.332S",
        "StartTimeInSeconds": 48.332
      }
    ],
    "vastAdId": ""
  }
],
"startTime": "PT46.47S",
"StartTimeInSeconds": 46.47
}
]
}

```

À la GetTracking demande suivante, MediaTailor répond avec la NextToken valeur : JF57ITe48t1441mv7TmLKuZLroxDzflslp6BiSNL1IjnzPVMDN0lqrBYycgMbKEb.

MediaTailor répond avec des publicités et des balises correspondant à StartTimeInSeconds celles définies lors NextToken de l'appel précédent.

Supposons que la réponse inclut désormais une autre annonce portant l'ID 9235407 en plus de l'annonce précédente portant l'ID 9935407. Les balises de l'ID d'annonce 9235407 portent les numéros 132,41 et 70,339StartTimeInSeconds.

MediaTailor effectue une itération sur toutes les balises de la session pour sélectionner celles dont la durée est StartTimeInSeconds supérieure à 52,286 secondes, à savoir les balises 3 et 4 de l'annonce portant l'ID 9235407 :

```

{
  "NextToken": ZkfknbvbsdgfbbsDFRdffg12EdffecFRvhjyjfhdhfnjtsG5SDGN
  "avails": [
    {
      "ads": [
        {
          "adId": "9235407",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "",
          "creativeSequence": "",
          "duration": "PT15.816S",

```

```

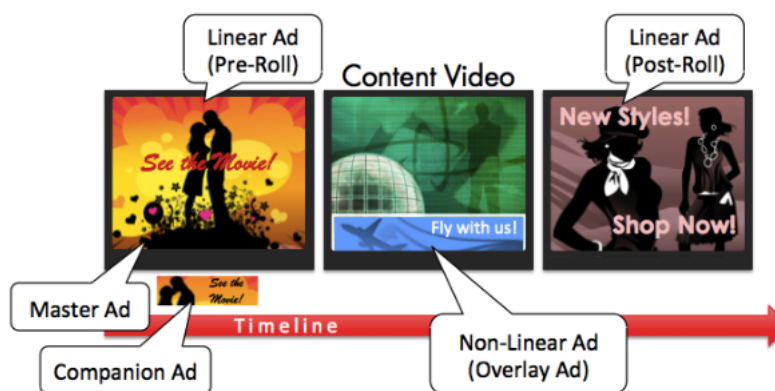
    "durationInSeconds": 19.716,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "startTime": "PT2M0S",
    "StartTimeInSeconds": 120.0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "http://adserver.com/tracking?event=complete"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0,
        "eventId": "8935414",
        "eventType": "firstQuartile",
        "startTime": "PT1M10.330S",
        "StartTimeInSeconds": 70.339
      },
      {
        "beaconUrls": [
          "http://adserver.com/tracking?event=thirdQuartile"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0,
        "eventId": "8935412",
        "eventType": "secondQuartile",
        "startTime": "PT2M12.41S",
        "StartTimeInSeconds": 132.41
      }
    ],
    "vastAdId": ""
  },
],
"startTime": "PT36.47S",
"StartTimeInSeconds": 36.47
}
]
}

```

Superposer des publicités

Pour les flux de diffusion en direct dans lesquels vous souhaitez augmenter la monétisation sans interrompre l'expérience de visionnage avec des publicités intermédiaires, vous pouvez tirer parti de votre AWS Elemental MediaTailor intégration actuelle pour définir un format publicitaire rendu côté client. Ce type de publicité est connu sous le nom de publicités superposées. Les publicités superposées sont des publicités vidéo non linéaires qui apparaissent sous la forme de « publicités », de « L-band publicités vidéo non linéaires », de « publicités illustrées », de « superpositions animées », de « publicités intégrées au contenu » ou de « publicités encadrées ».

MediaTailor détecte un SCTE-35 marqueur de type segmentation en `id=0x38` tant que signal intra-bande pour une opportunité d'insertion publicitaire en superposition. Le SCTE-35 marqueur envoie une demande MediaTailor au serveur de décision publicitaire (ADS), qui répond ensuite avec une charge utile publicitaire non linéaire dans la réponse VAST. MediaTailor analyse la réponse VAST afin de prendre en charge l'insertion d'annonces par superposition. MediaTailor n'assemble pas de publicités linéaires, mais indique au joueur qu'une publicité non linéaire est disponible pour être diffusée. Cette signalisation permet au joueur de récupérer et de corréliser les publicités non linéaires à diffuser à partir du point de suivi côté client. Le joueur gère ensuite l'affichage, les rapports et les autres tâches liées à ces publicités. Par exemple, le développeur du lecteur peut utiliser un SDK pour appareil fourni par un fournisseur qui prend en charge les formats publicitaires superposés. Pour plus d'informations sur les intégrations de suivi côté client, consultez [Client-side intégrations de suivi des publicités](#)



Rubriques

- [Conditions préalables à l'utilisation d'annonces incrustées avec MediaTailor](#)
- [Commencer à utiliser les publicités incrustées avec MediaTailor](#)
- [Enregistrement et statistiques pour les annonces incrustées dans MediaTailor](#)

- [Facturation des publicités incrustées dans MediaTailor](#)

Conditions préalables à l'utilisation d'annonces incrustées avec MediaTailor

Les conditions préalables suivantes s'appliquent lors de l'utilisation d'annonces en incrustation avec : MediaTailor

- Le flux de travail doit être en direct et non en vidéo à la demande (VOD).
- La réponse du serveur ADS (Ad Decision Server) doit être configurée pour renvoyer uniquement des publicités non linéaires dans la réponse VAST. MediaTailor ignore les publicités linéaires aux fins de l'assemblage des publicités.
- Le manifeste doit utiliser un message de signal SCTE-35 horaire de type segmentation `id=0x38` pour invoquer la fonctionnalité de superposition.
- Le fournisseur de streaming doit contrôler l'application client-appareil et être intégré à l'API de suivi côté MediaTailor client.

Commencer à utiliser les publicités incrustées avec MediaTailor

Cette section explique comment commencer à utiliser la fonctionnalité de superposition de MediaTailor. Vous allez configurer la signalisation SCTE-35, configurer les réponses du serveur ADS (Ad Decision Server) et configurer le contrôle au niveau de la session.

Rubriques

- [Activation des publicités superposées](#)
- [Suivi des publicités superposées à l'aide de métadonnées côté client](#)

Activation des publicités superposées

MediaTailor la prise en charge des annonces incrustées est activée par défaut. Un type de marqueur publicitaire SCTE-35 spécifique dans le manifeste déclenche l'insertion d'une publicité superposée. Certains joueurs étant susceptibles de ne pas prendre en charge le rendu des publicités incrustées côté client, vous pouvez désactiver cette fonctionnalité au niveau de la session.

Pour désactiver la prise en charge des publicités en superposition à l'aide des préfixes de lecture HLS ou DASH :

- Depuis le lecteur, initialisez une nouvelle session de MediaTailor lecture à l'aide d'une requête dans l'un des formats suivants, conformément à votre protocole :

- Exemple : format HLS

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

- Exemple : format DASH

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

Pour désactiver la prise en charge des publicités superposées à l'aide du préfixe d'initialisation de session :

- Sur le lecteur, créez un corps de message JSON pour la demande d'initialisation de session afin de MediaTailor :
 - Pour désactiver la prise en charge de la superposition de publicités, ajoutez un `overlays` objet en tant que clé de niveau supérieur avec une valeur de `off`. La `overlays` valeur par défaut est `on`.
 - (Facultatif) Fournissez tous les paramètres qui sont MediaTailor ensuite transmis à l'ADS à l'intérieur d'un `adsParams` objet. Ces paramètres correspondent aux paramètres `[player_params.param]` dans l'URL modèle d'ADS de la configuration de MediaTailor.

Exemple HLS :

```
POST master.m3u8  
{  
  "adsParams": {  
    "deviceType": "ipad"  
  },  
  "overlayAvails": "off"  
}
```

Exemple DASH :

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "androidmobile"
  },
  "overlayAvails": "off"
}
```

Signalisation manifeste

MediaTailor trigger overlay-ads support lorsqu'il voit un marqueur SCTE-35 spécifique dans le manifeste. Le signal requis est une commande d'épissage de type 6, ou signal horaire, qui est un signal Provider Overlay Advertisement Start. Ce signal possède un identifiant de type de segmentation de 0x38

L'exemple suivant montre le marqueur 0x38 SCTE-35 dans un objet JSON.

```
{
  "tableId": 252,
  "selectionSyntaxIndicator": false,
  "privateIndicator": false,
  "sectionLength": 53,
  "protocolVersion": 0,
  "encryptedPacket": false,
  "encryptedAlgorithm": 0,
  "ptsAdjustment": 0,
  "cwIndex": 0,
  "tier": 4095,
  "spliceCommandLength": 5,
  "spliceCommandType": 6,
  "spliceCommand": {
    "specified": true,
    "pts": 1800392
  },
  "descriptorLoopLength": 31,
  "descriptors": [
    {
      "spliceDescriptorTag": 2,
      "descriptorLength": 29,
      "identifier": "CUEI",
      "segmentationEventId": 158389361,
```

```

"segmentationEventCancelIndicator": false,
"programSegmentationFlag": true,
"segmentationDurationFlag": true,
"deliveryNotRestrictedFlag": false,
"webDeliveryAllowedFlag": true,
"noRegionalBlackoutFlag": true,
"archiveAllowedFlag": true,
"deviceResctrictions": 3,
"segmentationDuration": 1350000,
"segmentationUpidType": 9,
"segmentationUpidLength": 7,
"segmentationUpid": {
  "0": 111,
  "1": 118,
  "2": 101,
  "3": 114,
  "4": 108,
  "5": 97,
  "6": 121
},
"segmentationTypeId": 56,
"segmentNum": 1,
"segmentsExpected": 0
}
],
"crc": 2510422713
}

```

L'exemple suivant montre le signal SCTE-35 représenté sous forme de valeur binaire (base 32/hexadécimal) :

```
0xfc303500000000000000ff00506fe001b78c8001f021d435545490970d4717fd0000dbba009076f7665726c617
```

Les exemples suivants montrent le marqueur SCTE-35 dans les manifestes HLS et DASH.

Example: manifeste HLS

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:419
#EXT-X-DISCONTINUITY-SEQUENCE:3
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:09.231Z
```

```
#EXTINF:6.02,  
https://aws.cloudfront.net/media/asset1/index1_00007.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-KEY:METHOD=NONE  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:15.251Z  
#EXTINF:6.0,  
https://aws.cloudfront.net/media/asset1/index1_00001.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:21.251Z  
#EXTINF:4.0,  
https://aws.cloudfront.net/media/asset1/index1_00002.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-DATERANGE:ID="1692073825251-30-1", START-  
DATE="2023-08-15T04:30:25.251Z", DURATION=10.0, PLANNED-DURATION=10.0, SCTE35-  
OUT=0xfc303500000000000000ffff00506fe001b78c8001f021d435545490970d4717fdf00000dbba009076f7665726  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:25.251Z  
#EXTINF:2.0,  
https://aws.cloudfront.net/media/asset1/index1_00003.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:27.251Z  
#EXTINF:6.0,  
https://aws.cloudfront.net/media/asset1/index1_00004.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:33.251Z  
#EXTINF:2.0,  
https://aws.cloudfront.net/media/asset1/index1_00005.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:35.251Z  
#EXTINF:4.0,  
https://aws.cloudfront.net/media/asset1/index1_00006.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:39.251Z  
#EXTINF:6.02,  
https://aws.cloudfront.net/media/asset1/index1_00007.ts
```

Example: manifeste DASH

```
<?xml version="1.0"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  availabilityStartTime="2023-08-15T16:34:05.911Z" minBufferTime="PT30S"
  minimumUpdatePeriod="PT2S" profiles="urn:mpeg:dash:profile:isoff-live:2011"
  publishTime="2023-08-15T16:34:17.950Z" suggestedPresentationDelay="PT20S"
  timeShiftBufferDepth="PT1M30S" type="dynamic"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/ittf/
PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <Period xmlns="urn:mpeg:dash:schema:mpd:2011" id="1692117245944_1" start="PT0.033S">
```



```

<BaseUrl>https://aws.cloudfront.net/out/v1/abc/123/def/</BaseUrl>
<EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="90000">
  <Event duration="900000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
      <scte35:TimeSignal>
        <scte35:SpliceTime ptsTime="0"/>
      </scte35:TimeSignal>
      <scte35:SegmentationDescriptor segmentNum="0" segmentationDuration="900000"
segmentationEventCancelIndicator="false" segmentationEventId="1"
segmentationTypeId="56" segmentsExpected="0" subSegmentNum="0"
subSegmentsExpected="0">
        <scte35:SegmentationUpid segmentationUpidFormat="hexBinary"
segmentationUpidType="14">63736f7665726c6179</scte35:SegmentationUpid>
      </scte35:SegmentationDescriptor>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
<AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
  <Representation bandwidth="3000000" codecs="avc1.4D4028" frameRate="30/1"
height="1080" id="1" width="1920">
    <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_1_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_1_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
      <SegmentTimeline>
        <S d="60000" r="6" t="1000"/>
        <S d="30000" t="421000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
  <Representation bandwidth="2499968" codecs="avc1.4D4028" frameRate="30/1"
height="1080" id="2" width="1920">
    <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_2_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_2_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
      <SegmentTimeline>
        <S d="60000" r="6" t="1000"/>
        <S d="30000" t="421000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>

```

```

        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2200000" codecs="avc1.4D401F" frameRate="30/1"
height="720" id="3" width="1280">
        <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_3_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_3_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
            <SegmentTimeline>
                <S d="60000" r="6" t="1000"/>
                <S d="30000" t="421000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <Label>Alternate Audio</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000" codecs="mp4a.40.2"
id="9">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_audio_9_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_audio_9_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="48000">
            <SegmentTimeline>
                <S d="98304" t="0"/>
                <S d="96256" t="98304"/>
                <S d="95232" t="194560"/>
                <S d="96256" r="2" t="289792"/>
                <S d="95232" t="578560"/>
                <S d="46080" t="673792"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
</MPD>

```

Réponse du serveur ADS (Ad Decision Server)

La réponse ADS doit contenir un événement de suivi valide. Au minimum, l'événement de suivi peut être un événement `Impression` de suivi. L'événement de suivi doit contenir au moins une `NonLinear` annonce. Cette annonce est une publicité superposée, qui prend la forme d'une ressource statique, HTML ou `iFrame`.

```
<vmap AdBreak breaktype="linear" breakId="csoverlay"
```

Si la réponse VAST est un VMAP avec `breakType=nonlinear`, les métadonnées `Avail` se trouvent dans l'objet `nonLinearAvails` racine. Si la réponse VAST est une réponse VMAP avec un `breakType=linear`, ou une réponse VAST simple sans VMAP, les métadonnées `Avail` se trouvent dans l'objet `avails` racine.

La réponse VAST suivante est une réponse VMAP encapsulée avec une `breakType` valeur `linear`.

Outre la réponse VMAP encapsulée, prend MediaTailor également en charge une réponse VMAP encapsulée avec une `breakType` valeur de `nonlinear` et une réponse VAST simple.

```
<?xml version="1.0" encoding="utf-8"?>
<vmap:VMAP xmlns:vmap="http://www.iab.net/vmap-1.0" version="1.0">
  <vmap:AdBreak breakType="linear" breakId="csoverlay">
    <vmap:AdSource allowMultipleAds="true" followRedirects="true" id="1">
      <vmap:VASTAdData>
        <VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="3.0"
xsi:noNamespaceSchemaLocation="vast.xsd">
          <Ad sequence="1">
            <InLine>
              <AdSystem>2.0</AdSystem>
              <AdTitle>2</AdTitle>
              <Impression><![CDATA[https://adserver.com/beacon=impression]]></
Impression>
              <Creatives>
                <Creative>
                  <NonLinearAds>
                    <NonLinear width="640" height="360" id="18">
                      <StaticResource creativeType="text/js_ref"><![CDATA[https://
client-side-ads.com/tags/static/ctv-generic/overlay001.json?iv_geo_country%3DUS%26]]></
StaticResource>
                    </NonLinear>
```

```

        </NonLinearAds>
      </Creative>
    </Creatives>
  </Inline>
</Ad>
</VAST>
</vmap:VASTAdData>
</vmap:AdSource>
<vmap:TrackingEvents>
  <vmap:Tracking event="breakStart"><![CDATA[https://adserver.com/
beacon=breakstartimpression]]></vmap:Tracking>
  <vmap:Tracking event="breakEnd"><![CDATA[https://adserver.com/
beacon=breakendimpression]]></vmap:Tracking>
</vmap:TrackingEvents>
</vmap:AdBreak>
</vmap:VMAP>

```

Exemple 1 : source du manifeste DASH pour MediaTailor

```

<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"
  availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2014:xml+bin">
      <Event duration="540000" id="144">
        <scte35:Signal>
          <scte35:Binary>SCTE35-binary</scte35:Binary>
        </scte35:Signal>
      </Event>
    </EventStream>
    ...
  </Period>
  <Period start="PT46921.928S" id="49">
    ...

```

```
</Period>
</MPD>
```

Exemple 2 : manifeste DASH MediaTailor personnalisé contenant une décoration d'identifiant publicitaire

```
<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"
  availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling" timescale="90000">
      <Event presentationTime="13500000" duration="1351350">
        <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "adId","ad_position":
"adId", "ad_type":"overlay","creative_id": "creativeId","tracking_uri":
"trackingUri"}]]]></Event>
      </EventStream>
      ...
    </Period>
    <Period start="PT46921.928S" id="49">
      ...
    </Period>
  </MPD>
```

Suivi des publicités superposées à l'aide de métadonnées côté client

MediaTailor place les publicités superposées dans `nonLinearAdsList` le catalogue. L'API de suivi MediaTailor côté client possède deux objets racines, appelés `avails` et `nonLinearAvails`. Si la réponse VAST est un VMAP avec `breakType` `nonlinear`, les métadonnées `Avail` se trouvent dans l'objet `nonLinearAvails` racine. Si la réponse VAST est une réponse VMAP avec un `breakType` `linear`, ou une réponse VAST simple sans VMAP, les métadonnées `Avail` se trouvent dans l'`avails` objet racine.

Pour plus d'informations sur le suivi côté client, consultez. [Client-side suivi des publicités](#)

L'exemple suivant montre une réponse VAST simple ou une réponse VMAP avec une breakType valeur delinear.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakstartimpression"
          ],
          "eventType": "breakStart"
        },
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakendimpression"
          ],
          "eventType": "breakEnd"
        }
      ],
      "adMarkerDuration": null,
      "ads": [],
      "availId": "828",
      "availProgramDateTime": null,
      "duration": "PT0S",
      "durationInSeconds": 0,
      "meta": null,
      "nonLinearAdsList": [
        {
          "extensions": null,
          "nonLinearAdList": [
            {
              "adId": "",
              "adParameters": null,
              "adSystem": "2.0",
              "adTitle": "2",
              "apiFramework": null,
              "clickThrough": null,
              "clickTracking": null,
              "clickTrackingId": null,
              "creativeAdId": "",
              "creativeId": "18",
```

```

        "creativeSequence": "",
        "duration": null,
        "durationInSeconds": 0,
        "expandedHeight": null,
        "expandedWidth": null,
        "height": "360",
        "htmlResource": null,
        "iFrameResource": null,
        "maintainAspectRatio": false,
        "minSuggestedDuration": null,
        "scalable": false,
        "staticResource": "https://client-side-ads.com/tags/static/ctv-generic/overlay001.json?iv_geo_country%3DUS%26",
        "staticResourceCreativeType": "text/js_ref",
        "width": "640"
    },
    ],
    "trackingEvents": [
        {
            "beaconUrls": [
                "https://adserver.com/beacon=impression"
            ],
            "duration": null,
            "durationInSeconds": 0,
            "eventId": null,
            "eventProgramDateTime": null,
            "eventType": "impression",
            "startTime": null,
            "startTimeInSeconds": 0
        }
    ]
},
{
    "startTime": "PT1M46.08S",
    "startTimeInSeconds": 106.08
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": null,
"nonLinearAvails": []
}

```

L'exemple suivant montre une réponse VMAP simple avec une breakType valeur de nonLinear.

```
{
  "avails": [],
  "dashAvailabilityStartTime": null,
  "hlsAnchorMediaSequenceNumber": null,
  "nextToken": null,
  "nonLinearAvails": [
    {
      "adBreakTrackingEvents": [
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakstartimpression"
          ],
          "eventType": "breakStart"
        },
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakendimpression"
          ],
          "eventType": "breakEnd"
        }
      ],
      "adMarkerDuration": null,
      "ads": [],
      "availId": "828",
      "availProgramDateTime": null,
      "duration": "PT0S",
      "durationInSeconds": 0,
      "meta": null,
      "nonLinearAdsList": [
        {
          "extensions": null,
          "nonLinearAdList": [
            {
              "adId": "",
              "adParameters": null,
              "adSystem": "2.0",
              "adTitle": "2",
              "apiFramework": null,
              "clickThrough": null,
              "clickTracking": null,
              "clickTrackingId": null,
              "creativeAdId": "",

```



```

        "creativeId": "18",
        "creativeSequence": "",
        "duration": null,
        "durationInSeconds": 0,
        "expandedHeight": null,
        "expandedWidth": null,
        "height": "360",
        "htmlResource": null,
        "iFrameResource": null,
        "maintainAspectRatio": false,
        "minSuggestedDuration": null,
        "scalable": false,
        "staticResource": "https://client-side-ads.com/tags/static/ctv-generic/overlay001.json?iv_geo_country%3DUS%26",
        "staticResourceCreativeType": "text/js_ref",
        "width": "640"
    },
    "trackingEvents": [
        {
            "beaconUrls": [
                "https://adserver.com/beacon=impression"
            ],
            "duration": null,
            "durationInSeconds": 0,
            "eventId": null,
            "eventProgramDateTime": null,
            "eventType": "impression",
            "startTime": null,
            "startTimeInSeconds": 0
        }
    ]
},
"startTime": "PT1M46.08S",
"startTimeInSeconds": 106.08
}
]
}

```

Enregistrement et statistiques pour les annonces incrustées dans MediaTailor

Cette section explique la journalisation et les statistiques relatives aux annonces incrustées.

MediaTailor Pour plus d'informations sur la configuration de la journalisation, consultez [Surveillance et balisage AWS Elemental MediaTailor resources](#).

Rubriques

- [CloudWatch journaux](#)
- [CloudWatch métriques](#)

CloudWatch journaux

CloudWatch collecte les informations de journal suivantes concernant les publicités incrustées :

- VAST_RESPONSE- Affiche des informations sur la liste d'annonces non linéaires.
- FILLED_PROVIDER_OVERLAY- Affiche des informations sur les publicités non linéaires.

Note

RAW_ADS_RESPONSE Il s'agit d'un événement facultatif qui montre la réponse initiale de l'ADS. L'utilisation de cet événement est particulièrement utile dans un environnement de préparation et de test. Pour activer cet événement sur une configuration ou un compte, envoyez un ticket au AWS Support.

CloudWatch métriques

MediaTailor collecte les statistiques relatives aux annonces superposées séparément des autres statistiques ADS. MediaTailor collecte ces statistiques après avoir récupéré avec succès les publicités de l'ADS. Il n'est pas nécessaire d'interroger l'GetTrackingAPI pour collecter les métriques.

Le tableau suivant décrit les CloudWatch statistiques relatives aux annonces incrustées :

Métrique	Description
<code>AdDecisionServer.OverlayAds</code>	Le nombre d'annonces incrustées incluses dans les réponses ADS au cours de la CloudWatch période que vous avez spécifiée.
<code>AdDecisionServer.OverlayErrors</code>	Le nombre de réponses par code d'état autres que HTTP, de réponses vides et de réponses périmées MediaTailor reçues de l'ADS au cours de la CloudWatch période que vous avez spécifiée.
<code>AdDecisionServer.OverlayFilled</code>	Le nombre d'offres qui ont été remplies avec succès avec au moins une annonce superposée : • 1 - Il existe au moins une annonce valide. • 0 - Soit MediaTailor n'a reçu aucune publicité superposée, soit il y a eu un autre échec. <code>SampleCount</code> suit le nombre de fichiers disponibles. <code>Sums</code> suit le nombre de versions de superposition remplies avec succès.
<code>AdDecisionServer.OverlayMinSuggestedDuration</code>	Somme des <code>minSuggestedDuration</code> durées, en millisecondes, de toutes les publicités MediaTailor reçues de l'ADS pendant la CloudWatch période que vous avez spécifiée. Si <code>minSuggestedDuration</code> n'est pas spécifié, la durée affichée est la durée planifiée.
<code>AdDecisionServer.OverlayLatency</code>	Le temps de réponse, en millisecondes, pour les demandes adressées à l' MediaTailor ADS.

Métrique	Description
<code>AdDecisionServer.OverlayTimeouts</code>	Le nombre de demandes d'expiration adressées à l'ADS au cours de la CloudWatch période que vous avez spécifiée.
<code>AdsBilled</code>	Pour plus d'informations sur les publicités facturées, consultez Facturation des publicités incrustées dans MediaTailor .
<code>Avail.*</code>	Parce qu' MediaTailor il ne planifie pas les publicités incrustées et CloudWatch n'affiche aucun indicateur <code>Avail.X</code> .
<code>SkippedReason.*</code>	Parce qu' MediaTailor il ne planifie pas les publicités incrustées et CloudWatch n'affiche aucun indicateur <code>SkippedReason.X</code> .

Facturation des publicités incrustées dans MediaTailor

MediaTailor facture les clients en fonction du nombre d'annonces non linéaires figurant dans la réponse ADS. Ce chiffre inclut les publicités non linéaires qui s'étendent au-delà de la durée de la pause. Une fois le fichier MediaTailor rempli, il facture les annonces qu'il a publiées.

Pour les flux de travail de prélecture, MediaTailor ne facture pas les publicités lors de la récupération du préchargement, mais plutôt lorsqu'une publicité compatible apparaît dans la fenêtre de consommation de cette session.

Pour des informations de facturation supplémentaires, consultez <https://aws.amazon.com/mediatailor/pricing/>.

Décoration d'identification publicitaire

AWS Elemental MediaTailor effectue l'assemblage des publicités côté serveur lors de la transition entre le contenu et les pauses publicitaires. MediaTailor peut conditionner le manifeste à l'aide de métadonnées associées aux publicités qui ont été assemblées. Cela peut apporter les avantages suivants :

Important

La décoration de l'identifiant publicitaire n'est pas compatible avec les méthodes d'insertion de publicités guidées par le serveur (SGAI), car les champs qui remplissent les X-AD-CREATIVE-SIGNALING en-têtes ne sont connus que lorsque la liste des ressources est extraite pendant la diffusion de la publicité, et non lorsque le manifeste pouvant être mis en cache est initialement écrit.

- L'heure de début de la vidéo (VST) s'améliore
- MediaTailor peut prendre en charge un modèle hybride d'insertion publicitaire côté serveur et d'insertion publicitaire guidée par le serveur
- Les sessions côté serveur peuvent créer des chronologies de diffusion à l'aide de marqueurs de position des annonces
- Pour les sessions côté client qui créent déjà des chronologies de lecture avec l' MediaTailor API, la session VST s'améliore, car la session ne repose pas sur l'appel de l'API de suivi pour créer la chronologie
- Il est possible d'en tirer parti MediaTailor pour l'insertion de publicités côté serveur ainsi que pour les publicités affichées sur scène côté client. Ainsi, le kit de développement logiciel (SDK) d'un joueur n'a pas besoin d'une intégration distincte pour appeler directement les entités diffusant des publicités côté client. MediaTailor peut vendre les publicités via le manifeste et l'API de suivi côté client.

Il existe des normes pour associer chaque ressource publicitaire créative à un identifiant unique. Cette association permet aux annonceurs, aux agences, aux fournisseurs et aux éditeurs de relier un actif publicitaire créatif à leurs flux de travail indépendants. À mesure que les indicateurs et la surveillance des flux continuent de s'améliorer et que de plus en plus de distributeurs utilisent des architectures d'insertion basées sur des serveurs, il devient nécessaire de communiquer avec précision les identifiants attribués aux actifs créatifs individuels dans une interleaved/stitched présentation, par exemple dans le manifeste personnalisé.

Rubriques

- [Activation de la signalisation des identifiants publicitaires pour les sessions](#)
- [Insertion de manifestes et de métadonnées publicitaires](#)
- [Interactions avec le serveur de décision publicitaire \(ADS\)](#)

- [API de suivi côté client](#)

Activation de la signalisation des identifiants publicitaires pour les sessions

La fonction de signalisation de l'identifiant publicitaire doit être activée lors de l'initialisation de la session. Le processus d'activation de cette fonctionnalité est différent de la création de sessions utilisant le préfixe de HLS/DASH lecture (initialisation implicite de session) ou du préfixe d'initialisation de session (initialisation explicite de session).

Pour activer l'identifiant publicitaire pour la session à l'aide des HLS/DASH préfixes de lecture

- Depuis le lecteur, initialisez une nouvelle session de MediaTailor lecture à l'aide d'une requête dans l'un des formats suivants, conformément à votre protocole :

- Exemple : format HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.adSignalingEnabled=true
```

- Exemple : format DASH

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.adSignalingEnabled=true
```

Pour activer l'identifiant publicitaire pour la session à l'aide du préfixe d'initialisation de session

- Sur le lecteur, créez un corps de message JSON pour la demande d'initialisation de session afin de MediaTailor :
 - À l'intérieur d'un `adsParams` objet, fournissez tous les paramètres que MediaTailor doit être transmis à l'ADS. Ces paramètres correspondent aux `[player_params.param]` paramètres de l'URL du modèle ADS de la MediaTailor configuration.
 - Pour activer la signalisation par identifiant publicitaire, ajoutez un `adSignaling` objet en tant qu'objet de niveau supérieur, puis ajoutez un paramètre appelé `enabled` et une valeur `true`. La `adSignaling` valeur par défaut est `disabled`.
- Exemple : format HLS

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "adSignaling": {
    "enabled": "true"
  },
  "reportingMode": "client"
}
```

- Exemple : format DASH

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "adSignaling": {
    "enabled": "true"
  },
  "reportingMode": "client"
}
```

Insertion de manifestes et de métadonnées publicitaires

Au cours du processus d'assemblage des publicités, MediaTailor ajoute au manifeste l'identifiant unique associé à chaque création en cours de montage. MediaTailor obtient l'identifiant unique de la création à partir de la valeur d'`id`attribut de cette création dans la réponse VAST. Si la création ne possède pas de valeur d'attribut ID, elle MediaTailor publiera une valeur vide (`id=""`).

MediaTailor utilise un signal de métadonnées intégré au manifeste pour dissocier les dépendances entre l'API de suivi client pour les métadonnées de création publicitaire et timing/positioning dans le calendrier global. Ce découplage réduit la latence de lecture (en particulier dans les scénarios de VOD), où l'interface utilisateur (UI) du lecteur affiche les positions des pauses publicitaires dans la chronologie avant d'initialiser la lecture.

Les métadonnées ajoutées prennent les formes suivantes :

- Pour les manifestes HLS, les métadonnées ajoutées prennent la forme de DATERANGE balises pour chaque annonce pendant la période de disponibilité.
- Pour les manifestes DASH, les métadonnées ajoutées prennent la forme d'un Event élément au sein de chaque période publicitaire.

Le corps du message JSON suivant montre un exemple de réponse VAST :

```
{
  "version": 1,
  "identifiers": [
    {
      "scheme": "urn:smp:ul:060E2B34.01040101.01200900.00000000",
      "value": "creativeId",
      "ad_position": "adId",
      "ad_type": "adType",
      "tracking_uri": "trackingUri",
      "custom_vast_data": "customVastData"
    }
  ]
}
```

Dans l'exemple précédent :

- *creativeId* est la valeur d'Id attribut de l'Creative élément de l'annonce
- *adId* est soit le numéro de séquence HLS associé au début de l'annonce, soit l'identifiant de période DASH de l'annonce
- *adType* est soit *overlay*, *avail* soit, sur la base de la réponse VAST
- *trackingUri* est le point de terminaison de suivi relatif pour la MediaTailor session, au format `../../../../tracking/hashed-account-id/origin-id/session-id`
- *customVastData* est une valeur MediaTailor extraite de l'extension `creative_signaling` VAST. MediaTailor utilise le contenu du nœud CDATA, s'il est présent. Consultez la [Interactions avec le serveur de décision publicitaire \(ADS\)](#) section pour plus de détails et un exemple de réponse VAST.

Personnalisation des manifestes HLS avec des métadonnées publicitaires

Pour un flux HLS en direct, ajoutez des métadonnées MediaTailor uniquement lorsque le flux contient des PROGRAM-DATA-TIME balises, au moins une fois par durée de manifeste. Pour un flux de

vidéo à la demande (VOD), MediaTailor ajoute PROGRAM-DATE-TIME au moins un segment dans le manifeste personnalisé, où l'heure de début de chaque ressource VOD est Epoch Zero (). 1970-01-01T00:00:00Z Si le manifeste d'origine possède un PROGRAM-DATE-TIME contenu existant, il MediaTailor préserve ce contenu.

MediaTailor personnalise le manifeste avec les créations renvoyées par le serveur de décision publicitaire (ADS). Pour chaque annonce, inclut MediaTailor également un DATERANGE tag qui couvre la durée de l'annonce. Le format des DATERANGE balises est similaire à celui décrit dans la section [Signalisation publicitaire créative dans DASH et HLS](#) de la version 2023 de la publication technique de la SVA.

Le DATERANGE qui MediaTailor génère possède des valeurs d'identification uniques. Pour garantir l'unicité (conformément aux directives spécifiées dans [Mapping SCTE-35 into EXT-X-DATERANGE](#)), MediaTailor associe le MEDIA-SEQUENCE numéro du premier segment publicitaire de l'annonce au numéro de séquence de l'annonce contenue dans la publication.

Pour les pauses publicitaires insuffisamment remplies dans les configurations dans lesquelles Slate est activée MediaTailor , ajoutez les segments de Slate à la fin de la publication, séparés par une DISCONTINUITY balise, mais sans aucune métadonnée. DATERANGE

Pour chaque annonce insérée dans le manifeste personnalisé, les métadonnées créatives sont MediaTailor ajoutées, représentées sous forme de données codées en base64 dans une balise personnalisée. DATERANGE

Exemple Origine HLS linéaire (**#EXT-X-CUE-OUT**) :

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:398
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:20:01.397Z
#EXTINF:6.006,
index_1_398.ts?m=1676054627
#EXTINF:5.873,
index_1_399.ts?m=1676054627
#EXT-OATCLS-SCTE35:/DA1AAAAAyiYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
#EXT-X-CUE-OUT:59.993
#EXTINF:6.139,
index_1_400.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=6.139,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
```

```
#EXTINF:6.006,  
index_1_401.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=12.145,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_402.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=18.151,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_403.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=24.157,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_404.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=30.163,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_405.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=36.169,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_406.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=42.175,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_407.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=48.181,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:6.006,  
index_1_408.ts?m=1676054627  
#EXT-X-CUE-OUT-CONT:ElapsedTime=54.187,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf  
+//jP197P4AUmNiAAEBAQAase4/gA==  
#EXTINF:5.806,  
index_1_409.ts?m=1676054627  
#EXT-X-CUE-IN  
#EXTINF:6.206,  
index_1_410.ts?m=1676054627  
#EXTINF:6.006,  
index_1_411.ts?m=1676054627  
#EXTINF:6.006,  
index_1_412.ts?m=1676054627
```

Exemple Origine HLS linéaire (#EXT-X-DATERANGE) :

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:25
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:53.389Z
#EXTINF:6.006,
index_1_25.ts?m=1676056675
#EXTINF:6.006,
index_1_26.ts?m=1676056675
#EXTINF:6.006,
index_1_27.ts?m=1676056675
#EXTINF:1.869,
index_1_28.ts?m=1676056675
#EXT-X-DATERANGE:ID="2",START-DATE="2023-02-10T19:20:13.276Z",PLANNED-
DURATION=59.993,SCITE35-
OUT=0xFC3025000000003289800FFF01405000000027FEFFF8CF97DECFE00526362000101010000B1EE3F80
#EXTINF:6.139,
index_1_29.ts?m=1676056675
#EXTINF:6.006,
index_1_30.ts?m=1676056675
#EXTINF:6.006,
index_1_31.ts?m=1676056675
#EXTINF:6.006,
index_1_32.ts?m=1676056675
#EXTINF:6.006,
index_1_33.ts?m=1676056675
#EXTINF:6.006,
index_1_34.ts?m=1676056675
#EXTINF:6.006,
index_1_35.ts?m=1676056675
#EXTINF:6.006,
index_1_36.ts?m=1676056675
#EXTINF:6.006,
index_1_37.ts?m=1676056675
#EXTINF:5.806,
index_1_38.ts?m=1676056675
#EXT-X-DATERANGE:ID="2",START-DATE="2023-02-10T19:20:13.276Z",END-
DATE="2023-02-10T19:21:13.269Z",DURATION=59.993
#EXTINF:6.206,
index_1_39.ts?m=1676056675
#EXTINF:6.006,
```

```
index_1_40.ts?m=1676056675
```

Exemple Manifeste personnalisé HLS linéaire (avec signalisation publicitaire créative) :

Le DATERANGE qui MediaTailor génère possède des valeurs d'identification uniques. Pour garantir l'unicité (conformément aux directives spécifiées dans [Mapping SCTE-35 into EXT-X-DATERANGE](#)), MediaTailor associe le MEDIA-SEQUENCE numéro du premier segment publicitaire de l'annonce au numéro de séquence de l'annonce contenue dans la publication.

Dans l'exemple suivant, MEDIA-SEQUENCE 421 est MediaTailor concatène avec le numéro de position de l'annonce.

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:418
#EXT-X-DISCONTINUITY-SEQUENCE:5
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_397.ts?m=1676054627
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_398.ts?m=1676054627
#EXTINF:5.873,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_399.ts?m=1676054627
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-1",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056813
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056814
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056815
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056816
```

```
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056817
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056818
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056819
#EXTINF:1.001,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056820
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-1",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXT-X-DATERANGE:ID="421-2",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056821
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056822
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056823
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056824
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056825
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056826
#EXTINF:2.002,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056827
#EXTINF:1.001,
.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056828
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
```

```

#EXT-X-DATERANGE:ID="421-2",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXT-X-DATERANGE:ID="421-3",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056829
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056830
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056831
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056832
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056833
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056834
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056835
#EXTINF:1.001,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056836
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-3",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=29.997
#EXT-X-DATERANGE:ID="421-4",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056837
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056838
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056839
#EXTINF:2.002,

```

```

.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056840
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056841
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056842
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056843
#EXTINF:1.001,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056844
#EXT-X-DISCONTINUITY
#EXT-X-DATERANGE:ID="421-4",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXTINF:6.206,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_410.ts?m=1676054627
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_411.ts?m=1676054627

```

Exemple Origine VOD HLS (avec signaux SCTE) :

```

#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:1
#EXT-X-PLAYLIST-TYPE:VOD
#EXTINF:6,
index_720p1500k_00001.ts
#EXTINF:6,
index_720p1500k_00002.ts
#EXTINF:6,
index_720p1500k_00003.ts
#EXTINF:6,
index_720p1500k_00004.ts
#EXTINF:6,
index_720p1500k_00005.ts
#EXT-X-CUE-OUT:0
#EXT-X-CUE-IN

```

```
#EXTINF:6,  
index_720p1500k_00006.ts  
#EXTINF:6,  
index_720p1500k_00007.ts  
#EXTINF:6,  
index_720p1500k_00008.ts  
#EXTINF:6,  
index_720p1500k_00009.ts  
#EXTINF:6,  
index_720p1500k_00010.ts  
#EXTINF:6,  
index_720p1500k_00011.ts  
#EXTINF:6,  
index_720p1500k_00012.ts
```

Exemple Origine de la VOD HLS :

```
#EXTM3U  
#EXT-X-VERSION:3  
#EXT-X-TARGETDURATION:7  
#EXT-X-MEDIA-SEQUENCE:1  
#EXT-X-PLAYLIST-TYPE:VOD  
#EXTINF:6,  
index_720p1500k_00001.ts  
#EXTINF:6,  
index_720p1500k_00002.ts  
#EXTINF:6,  
index_720p1500k_00003.ts  
#EXTINF:6,  
index_720p1500k_00004.ts  
#EXTINF:4,  
index_720p1500k_00005.ts  
#EXTINF:2,  
index_720p1500k_00006.ts  
#EXTINF:6,  
index_720p1500k_00007.ts  
#EXTINF:6,  
index_720p1500k_00008.ts  
#EXTINF:6,  
index_720p1500k_00009.ts  
#EXTINF:6,  
index_720p1500k_00010.ts  
#EXTINF:6,
```



```
index_720p1500k_00011.ts
#EXTINF:6,
index_720p1500k_00012.ts
```

Exemple Manifeste personnalisé VOD HLS :

MediaTailor PROGRAM-DATE-TIMEs'ajoute aux manifestes de VOD afin de les utiliser comme points d'ancrage pour les DATERANGE éléments HLS qui indiquent les positions des annonces.

Le DATERANGE qui MediaTailor génère possède des valeurs d'identification uniques. Pour garantir l'unicité (conformément aux directives spécifiées dans [Mapping SCTE-35 into EXT-X-DATERANGE](#)), MediaTailor associe le MEDIA-SEQUENCE numéro du premier segment publicitaire de l'annonce au numéro de séquence de l'annonce contenue dans la publication.

Dans l'exemple suivant, MEDIA-SEQUENCE 421 est MediaTailor concatène avec le numéro de position de l'annonce.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-PLAYLIST-TYPE:VOD
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:1
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXT-X-PROGRAM-DATE-TIME:1970-01-01T00:00:00Z
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00001.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00002.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00003.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00004.ts
#EXTINF:4.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00005.ts
#EXT-X-DISCONTINUITY
#EXTINF:2.002,
../../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/28
```

```
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/29
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/30
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/31
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/32
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/33
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/34
#EXTINF:1.001,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/35
#EXT-X-DISCONTINUITY
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/36
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/37
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/38
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/39
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/40
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/41
#EXTINF:2.002,
../.../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/42
#EXTINF:1.001,
```

```

../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/43
#EXT-X-DISCONTINUITY
#EXTINF:2.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00006.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00007.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00008.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00009.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00010.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00011.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00012.ts
#EXT-X-ENDLIST
#EXT-X-DATERANGE:ID="5-1",START-DATE="1970-01-01T00:00:28.000Z",END-
DATE="1970-01-01T00:00:43.015Z",DURATION=15.015
#EXT-X-DATERANGE:ID="5-2",START-DATE="1970-01-01T00:00:43.015Z",END-
DATE="1970-01-01T00:00:58.030Z",DURATION=15.01

```

Personnalisation des manifestes DASH avec des métadonnées publicitaires

MediaTailor personnalise le manifeste avec les créations renvoyées par le serveur de décision publicitaire (ADS). Pour chaque annonce, inclut MediaTailor également un `EventStream` élément qui couvre la durée de l'annonce. Le format des `Event` éléments est similaire à celui décrit dans la section [Signalisation créative publicitaire dans DASH et HLS](#) de la version 2023 de la publication technique SVA.

Pour les pauses publicitaires insuffisamment remplies sur les configurations dans lesquelles Slate MediaTailor est activée, ajoute la période Slate à la fin de la période de disponibilité, mais sans aucune métadonnée `EventStream`

Pour chaque annonce intégrée au manifeste personnalisé, les métadonnées créatives sont MediaTailor ajoutées, représentées sous la forme d'un CDATA élément au sein d'un Event élément.

Exemple Origine linéaire du DASH (attributs SCTE intégrés) :

```
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2023-02-10T21:08:40+00:00" minimumUpdatePeriod="PT6S"
  availabilityStartTime="2023-02-09T22:47:05.865000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT88.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT80141.456S" id="104" duration="PT304.103S">
    <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
    startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
    bitstreamSwitching="true">
      <SegmentTemplate timescale="60000" media="index_video_$RepresentationID$_0_
      $Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_0_init.mp4?
      m=1676062374" startNumber="151" presentationTimeOffset="4808487386">
        <SegmentTimeline>
          <S t="4824975858" d="360360" r="3"/>
          <S t="4826417298" d="316316"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="1" width="960" height="540" frameRate="30000/1001"
      bandwidth="1800000" codecs="avc1.4D401F"/>
      <Representation id="3" width="640" height="360" frameRate="30000/1001"
      bandwidth="1200000" codecs="avc1.4D401E"/>
      <Representation id="5" width="480" height="270" frameRate="30000/1001"
      bandwidth="800000" codecs="avc1.4D4015"/>
    </AdaptationSet>
    <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
    lang="eng">
      <Label>eng</Label>
      <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_0_
      $Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_0_init.mp4?
      m=1676062374" startNumber="151" presentationTimeOffset="3846790126">
        <SegmentTimeline>
          <S t="3859981294" d="287744"/>
          <S t="3860269038" d="288768"/>
          <S t="3860557806" d="287744"/>
          <S t="3860845550" d="288768"/>
          <S t="3861134318" d="252928"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </AdaptationSet>
  </Period>
</MPD>
```

```

    </SegmentTimeline>
  </SegmentTemplate>
  <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
  </Representation>
  <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
  </Representation>
  <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
  </Representation>
</AdaptationSet>
  <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>
<Period start="PT80445.560S" id="155" duration="PT44.978S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="4048044">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="207000"
tier="4095">
        <scte35:SpliceInsert spliceEventId="111" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
          <scte35:Program>
            <scte35:SpliceTime ptsTime="7239893422"/>
          </scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="4048044"/>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
  <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <SegmentTemplate timescale="60000" media="index_video_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_${RepresentationID$_0_init}.mp4?
m=1676062374" startNumber="156" presentationTimeOffset="4826733614">
      <SegmentTimeline>

```

```

        <S t="4826733614" d="284284"/>
        <S t="4827017898" d="360360" r="5"/>
        <S t="4829180058" d="252252"/>
    </SegmentTimeline>
</SegmentTemplate>
<Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
</AdaptationSet>
<AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
    <Label>eng</Label>
    <SegmentTemplate timescale="48000" media="index_audio_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_${RepresentationID$_0_init.mp4?
m=1676062374" startNumber="156" presentationTimeOffset="3861387246">
        <SegmentTimeline>
            <S t="3861387246" d="227328"/>
            <S t="3861614574" d="288768"/>
            <S t="3861903342" d="287744"/>
            <S t="3862191086" d="288768"/>
            <S t="3862479854" d="287744"/>
            <S t="3862767598" d="288768"/>
            <S t="3863056366" d="287744"/>
            <S t="3863344110" d="202752"/>
        </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>

```

```

    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:07:35.111Z"/>
  </Period>
  <Period start="PT80490.538S" id="163">
    <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
      <SegmentTemplate timescale="60000" media="index_video_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_${RepresentationID$_0_init.mp4?
m=1676062374" startNumber="164" presentationTimeOffset="4829432310">
        <SegmentTimeline>
          <S t="4829432310" d="348348"/>
          <S t="4829780658" d="360360" r="1"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
      <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
      <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
    </AdaptationSet>
    <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
      <Label>eng</Label>
      <SegmentTemplate timescale="48000" media="index_audio_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_${RepresentationID$_0_init.mp4?
m=1676062374" startNumber="164" presentationTimeOffset="3863546862">
        <SegmentTimeline>
          <S t="3863546862" d="278528"/>
          <S t="3863825390" d="287744"/>
          <S t="3864113134" d="288768"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
      </Representation>
      <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>

```

```

    </Representation>
    <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
      <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
  </AdaptationSet>
  <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:08:20.090Z"/>
</Period>
</MPD>

```

Exemple Manifeste personnalisé linéaire DASH (avec signalisation publicitaire créative) :

```

<MPD availabilityStartTime="2023-02-09T22:47:05.865000+00:00"
id="201" minBufferTime="PT10S" minimumUpdatePeriod="PT6S"
profiles="urn:mpeg:dash:profile:isoff-live:2011"
publishTime="2023-02-10T21:08:43+00:00" suggestedPresentationDelay="PT20.000S"
timeShiftBufferDepth="PT88.999S" type="dynamic" xmlns="urn:mpeg:dash:schema:mpd:2011"
xmlns:scte35="urn:scte:scte35:2013:xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://d3fch9e2fcarly.cloudfront.net/out/v1/
f9f38deca3f14fc4b5ab3cdbd76cfb9e/</BaseURL>
  <Location>https://777788889999.mediataylor.us-west-2.amazonaws.com/
v1/dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/out/v1/
f9f38deca3f14fc4b5ab3cdbd76cfb9e/index.mpd?
aws.sessionId=672ed481-4ffd-4270-936f-7c8403947f2e</Location>
  <Period duration="PT304.103S" id="104" start="PT80141.456S">
    <AdaptationSet bitstreamSwitching="true" id="1485523442" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
      <SegmentTemplate initialization="index_video_$RepresentationID$_0_init.mp4?
m=1676062374" media="index_video_$RepresentationID$_0_$Number$.mp4?m=1676062374"
presentationTimeOffset="4808487386" startNumber="151" timescale="60000">
        <SegmentTimeline>
          <S d="360360" r="3" t="4824975858"/>
          <S d="316316" t="4826417298"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation bandwidth="1800000" codecs="avc1.4D401F"
frameRate="30000/1001" height="540" id="1" width="960"/>
    </AdaptationSet>
  </Period>
</MPD>

```



```

        <Representation bandwidth="1200000" codecs="avc1.4D401E"
frameRate="30000/1001" height="360" id="3" width="640"/>
        <Representation bandwidth="800000" codecs="avc1.4D4015"
frameRate="30000/1001" height="270" id="5" width="480"/>
    </AdaptationSet>
    <AdaptationSet id="1377232898" lang="eng" mimeType="audio/mp4"
segmentAlignment="0">
        <Label>eng</Label>
        <SegmentTemplate initialization="index_audio_$RepresentationID$_0_init.mp4?
m=1676062374" media="index_audio_$RepresentationID$_0_$Number$.mp4?m=1676062374"
presentationTimeOffset="3846790126" startNumber="151" timescale="48000">
            <SegmentTimeline>
                <S d="287744" t="3859981294"/>
                <S d="288768" t="3860269038"/>
                <S d="287744" t="3860557806"/>
                <S d="288768" t="3860845550"/>
                <S d="252928" t="3861134318"/>
            </SegmentTimeline>
        </SegmentTemplate>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="4">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="6">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>
<Period id="155_1" start="PT22H20M45.56S">
    <BaseURL>https://777788889999.mediataylor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_1/</BaseURL>
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">

```

```

    <Event presentationTime="xxxxx" duration="1351350">
      <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smppte:ul:060E2B34.01040101.01200900.00000000","value": "155_1","ad_position":
"155_1", "ad_type":"avail","creative_id": "123","tracking_uri": "../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
    </Event>
  </EventStream>
  <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
  mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
  subsegmentAlignment="true" subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
  id="1" width="960">
      <SegmentTemplate initialization="asset_540_1_2init.mp4"
  media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
  id="3" width="640">
      <SegmentTemplate initialization="asset_360_1_1init.mp4"
  media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
  width="480">
      <SegmentTemplate initialization="asset_270_0_0init.mp4"
  media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
  <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">

```

```

        <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
        <Label>eng</Label>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
            <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
                <SegmentTimeline>
                    <S d="98304" t="0"/>
                    <S d="96256" r="1" t="98304"/>
                    <S d="95232" t="290816"/>
                    <S d="96256" r="2" t="386048"/>
                    <S d="48128" t="674816"/>
                </SegmentTimeline>
            </SegmentTemplate>
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
</Period>
<Period id="155_2" start="PT22H21M0.575S">
    <BaseURL>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_2/</BaseURL>
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
        <Event presentationTime="0" duration="1351350">
            <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "155_2","ad_position":
"155_2", "ad_type":"avail","creative_id": "234","tracking_uri": "../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
        </Event>
    </EventStream>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">
            <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>

```

```

        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
    <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
            <S d="180180" r="6" t="0"/>
            <S d="90090" t="1261260"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
width="480">
    <SegmentTemplate initialization="asset_270_0_0init.mp4"
media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
            <S d="180180" r="6" t="0"/>
            <S d="90090" t="1261260"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
        <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
            <SegmentTimeline>
                <S d="98304" t="0"/>
                <S d="96256" r="1" t="98304"/>
                <S d="95232" t="290816"/>
                <S d="96256" r="2" t="386048"/>
                <S d="48128" t="674816"/>
            </SegmentTimeline>
        </SegmentTemplate>
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
</AdaptationSet>

```

```

</Period>
<Period id="155_3" start="PT22H21M15.59S">
  <BaseUrl>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_3/</BaseUrl>
  <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
    <Event presentationTime="0" duration="1351350">
      <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smppte:ul:060E2B34.01040101.01200900.00000000","value": "155_3","ad_position":
"155_3", "ad_type":"avail","creative_id": "345","tracking_uri": "../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
    </Event>
  </EventStream>
  <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">
      <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_$_Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
      <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_$_Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
width="480">
      <SegmentTemplate initialization="asset_270_0_0init.mp4"
media="asset_270_0_0_$_Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>

```

```

        <S d="90090" t="1261260"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
  <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
  <Label>eng</Label>
  <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
    <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
      <SegmentTimeline>
        <S d="98304" t="0"/>
        <S d="96256" r="1" t="98304"/>
        <S d="95232" t="290816"/>
        <S d="96256" r="2" t="386048"/>
        <S d="48128" t="674816"/>
      </SegmentTimeline>
    </SegmentTemplate>
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
  </Representation>
</AdaptationSet>
</Period>
<Period id="163" start="PT80490.538S">
  <AdaptationSet bitstreamSwitching="true" id="1485523442" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
    <SegmentTemplate initialization="index_video_${RepresentationID$_0_init.mp4?
m=1676062374" media="index_video_${RepresentationID$_0_${Number$.mp4?m=1676062374"
presentationTimeOffset="4829432310" startNumber="164" timescale="60000">
      <SegmentTimeline>
        <S d="348348" t="4829432310"/>
        <S d="360360" r="1" t="4829780658"/>
      </SegmentTimeline>
    </SegmentTemplate>
    <Representation bandwidth="1800000" codecs="avc1.4D401F"
frameRate="30000/1001" height="540" id="1" width="960"/>
    <Representation bandwidth="1200000" codecs="avc1.4D401E"
frameRate="30000/1001" height="360" id="3" width="640"/>
    <Representation bandwidth="800000" codecs="avc1.4D4015"
frameRate="30000/1001" height="270" id="5" width="480"/>

```

```

    </AdaptationSet>
    <AdaptationSet id="1377232898" lang="eng" mimeType="audio/mp4"
segmentAlignment="0">
        <Label>eng</Label>
        <SegmentTemplate initialization="index_audio_$RepresentationID$_0_init.mp4?
m=1676062374" media="index_audio_$RepresentationID$_0_$Number$.mp4?m=1676062374"
presentationTimeOffset="3863546862" startNumber="164" timescale="48000">
            <SegmentTimeline>
                <S d="278528" t="3863546862"/>
                <S d="287744" t="3863825390"/>
                <S d="288768" t="3864113134"/>
            </SegmentTimeline>
        </SegmentTemplate>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="4">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="6">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:08:20.090Z"/>
</Period>
</MPD>

```

Interactions avec le serveur de décision publicitaire (ADS)

MediaTailor utilise la valeur de `id` l'attribut créatif issue de la réponse VAST comme valeur dans la signalisation de l'identifiant publicitaire. Si la valeur de `id` l'attribut est vide ou absente de la réponse VAST, MediaTailor place une valeur vide dans le signal d'identification publicitaire.

Exemple Réponse VASTE :

L'exemple de réponse VAST suivant inclut une valeur d'idattribut pour le linéaire Creative en ligne. MediaTailor extrait la valeur de l'Extensionélément VAST personnalisé et place cette valeur dans les métadonnées créatives du manifeste.

```
<?xml version="1.0" encoding="utf-8"?>
<VAST version="3.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Ad sequence="3">
    <InLine>
      <AdSystem>2.0</AdSystem>
      <AdTitle>AD-caribbean2-15</AdTitle>
      <Impression><![CDATA[https://n8ljfs0xxx.execute-api.us-
west-2.amazonaws.com/v1/impression]]></Impression>
      <Creatives>
        <Creative sequence="3" apiFramework="inLine" id="1234">
          <Linear>
            <Duration>00:00:15</Duration>
            <MediaFiles>
              <MediaFile id="00002" delivery="progressive" type="video/
mp4" width="1280" height="720"><![CDATA[https://d3re4i3vgppxxx.cloudfront.net/Media/
Bumpers/AD-caribbean2-15-HD.mp4]]></MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
      <Extensions>
        <Extension type="creative_signaling"><![CDATA[999999|
TVN1DDNpFTchtpRj,E5TfTtcYd5IEzvEt,ChA050HcvWRGFY6Zp5VSS1xUEJ2B9p8GGhQIDzIQkFeQC-
Ho67FR3P9qNa6khSAGKgAyAA]]></Extension>
      </Extensions>
    </InLine>
  </Ad>
</VAST>
```

API de suivi côté client

L'exemple suivant montre comment un SDK de lecteur lie les métadonnées publicitaires du manifeste aux données complètes des événements de suivi de la charge utile de réponse de suivi côté client avec `et.creativeId` `adId`

Example Message JSON :

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "ads": [
        {
          "adId": "5",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "2.0",
          "adTitle": "AD-caribbean2-15",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "1234",
          "creativeSequence": "2",
          "duration": "PT15S",
          "durationInSeconds": 15,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "skipOffset": null,
          "startTime": "PT30S",
          "startTimeInSeconds": 30,
          "trackingEvents": [
            {
              "beaconUrls": [
                "https://myServer/impression"
              ],
              "duration": "PT15S",
              "durationInSeconds": 15,
              "eventId": "5",
              "eventProgramDateTime": null,
              "eventType": "impression",
              "startTime": "PT30S",
              "startTimeInSeconds": 30
            }
          ],
          "vastAdId": ""
        }
      ]
    }
  ],
}
```

```
    "availId": "5",
    "availProgramDateTime": null,
    "duration": "PT15S",
    "durationInSeconds": 15,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT30S",
    "startTimeInSeconds": 30
  }
],
"nextToken": "UFQ1TTM0Ljk2N1NfMjAyMi0xMS0xOFQwNDozMzo1Mi4yNDUxOTdaXzE%3D",
"nonLinearAvails": []
}
```

Utilisation AWS Elemental MediaTailor pour créer des flux linéaires assemblés

AWS Elemental MediaTailor channel assembly est un service de manifeste uniquement qui vous permet de créer des chaînes de streaming linéaires en utilisant votre contenu de vidéo à la demande (VOD) existant mélangé à du contenu en direct. MediaTailor ne touche jamais à vos segments de contenu, qui sont diffusés directement depuis votre serveur d'origine. MediaTailor Récupère plutôt les manifestes depuis votre origine et les utilise pour créer une fenêtre de manifeste glissante en direct qui fait référence aux segments de contenu sous-jacents. L'assemblage des canaux permet de suivre des éléments tels que le numéro de séquence multimédia nécessaire pour une lecture fluide d'une ressource à l'autre. Les flux linéaires assemblés sont créés à faible coût de fonctionnement en utilisant le contenu VOD codé et packagé à plusieurs débits existants.

Vous pouvez facilement monétiser les flux linéaires d'assemblage de chaînes en insérant des pauses publicitaires dans vos programmes sans avoir à conditionner le contenu à l'aide SCTE-35 de marqueurs. Vous pouvez utiliser l'assemblage de canaux avec le service d'insertion d'annonces MediaTailor ou tout autre service d'insertion de publicités côté serveur.

Pour commencer à assembler des canaux, consultez [the section called “Débuter avec l'assemblage de MediaTailor canaux”](#).

Pour plus d'informations sur l'intégration de Channel Assembly à un réseau de diffusion de contenu (CDN), consultez [Créez des canaux MediaTailor linéaires avec assemblage de canaux et CDN](#).

Rubriques

- [Types de flux de travail d'assemblage de canaux](#)
- [Utilisation des emplacements sources](#)
- [Utilisation des chaînes](#)
- [Ajouter un programme à la programmation d'une chaîne](#)
- [Insérez des publicités et des pauses publicitaires personnalisées dans le flux d'une chaîne](#)
- [Décalage temporel de lecture d'une chaîne](#)

Types de flux de travail d'assemblage de canaux

AWS Elemental MediaTailor Channel Assembly prend en charge trois types de flux de travail pour créer des chaînes de streaming linéaires. Chaque type de flux de travail utilise une combinaison différente de contenu source pour répondre à des cas d'utilisation spécifiques. Cette rubrique décrit chaque type de flux de travail, indique quand l'utiliser et propose des liens vers les procédures de configuration détaillées.

Type de flux de travail	Description	Mode de lecture	Cas d'utilisation
VOD-to-live	Assemble des contenus VOD préenregistrés dans un flux linéaire en direct	Boucle ou linéaire	24/7 chaînes issues d'une bibliothèque de contenu
Live-to-live	Transfère les flux source en direct sous forme de programmes sur une chaîne programmée	Linéaire uniquement	Actualités, sports, programmation régionale
VOD + Live-to-live (Mixte)	Combine la VOD et les sources en direct dans un seul programme de chaînes	Linéaire uniquement	Chaînes proposant principalement du contenu VOD et incluant des événements programmés en direct

VOD-to-live

Une VOD-to-live chaîne crée un flux linéaire entièrement à partir de contenu VOD pré-encodé et préemballé. Le flux ressemble à une diffusion en direct avec une fenêtre manifeste coulissante en direct, mais utilise uniquement des ressources VOD. Vous n'avez pas besoin de SCTE-35 marqueurs dans votre contenu source, car vous définissez des pauses publicitaires lorsque vous programmez des programmes.

Quand l'utiliser

- Vous disposez d'une bibliothèque de ressources VOD et souhaitez créer une chaîne 24/7 linéaire.

- Vous souhaitez monétiser le contenu à l'aide de pauses publicitaires, mais vous n'avez pas de SCTE-35 marqueurs dans votre contenu source.
- Vous voulez un flux linéaire simple et peu coûteux qui ne nécessite pas d'infrastructure de codage en direct.

Comment ça marche

1. Préparez votre contenu : encodez et empaquetez vos ressources VOD. Hébergez-les sur un serveur d'origine (Amazon S3 ou toute autre source HTTP). AWS Elemental MediaPackage
2. Créez un emplacement source : enregistrez votre serveur d'origine auprès de MediaTailor.
3. Ajouter des sources de VOD : enregistrez chaque élément de contenu et définissez ses configurations de package et ses groupes de sources.
4. Créer une chaîne — Choisissez un mode de lecture (boucle ou linéaire) et configurez les sorties.
5. Programmer des programmes : ajoutez des programmes à la programmation de votre chaîne dans l'ordre dans lequel vous souhaitez qu'ils soient diffusés.
6. Configurer les pauses publicitaires — Ajoutez des pauses publicitaires à chaque programme. Connectez-vous à une configuration SSAI pour des publicités personnalisées.
7. Démarrez votre chaîne et configurez un CDN — Démarrez la chaîne. Vous pouvez éventuellement ajouter un CDN devant celui-ci pour des économies d'échelle et de coûts.

Choix d'un mode de lecture

VOD-to-live les canaux prennent en charge deux modes de lecture :

- Loop (niveau de base) — Les programmes sont lus l'un après l'autre dans une boucle infinie. Lorsque le dernier programme est terminé, la lecture revient au premier. Il s'agit de l'option la plus simple pour une 24/7 chaîne. Le mode boucle ne prend pas en charge les sources en direct ni la liste de remplissage.
- Linéaire (niveau standard) — Les programmes sont lus une fois de suite. Le mode linéaire prend en charge la planification en fonction du temps et permet de combler les écarts entre les programmes.

Exigences relatives au contenu

Pour des transitions fluides, toutes les ressources VOD d'une même chaîne doivent disposer des éléments suivants :

- Le même nombre de rendus (vidéo, audio, sous-titres, etc.) pour toutes les sources d'un groupe de sources
- Durations de segment cohérentes pour tous les actifs. Nous recommandons une durée de 2 à 6 secondes.
- Débits et résolutions correspondants pour les rendus correspondants

Important

Channel Assembly ne prend pas en charge le codage par titre ni le codage ABR automatique. Ces méthodes produisent des actifs avec des nombres et des durées de rendu différents, ce qui entraîne des échecs de lecture lors des transitions de programme.

Pauses publicitaires et monétisation

Vous définissez des pauses publicitaires lorsque vous ajoutez des programmes à votre planning. Chaque pause publicitaire nécessite un décalage en millisecondes à partir du début du programme. Le décalage doit être aligné sur la limite d'un segment sur toutes les pistes de la source VOD.

Votre manifeste source de VOD peut contenir des marqueurs publicitaires. Pour HLS, il s'agit de EXT-X-CUE-OUT balises EXT-X-DATERANGE ou avec une durée. Pour DASH, utilisez `EventStream`. Lorsque ces marqueurs sont MediaTailor détectés, ils affichent leurs décalages sous forme d'options lorsque vous configurez le break publicitaire. Cela permet de placer les annonces de la manière la plus précise possible.

Pour diffuser des publicités personnalisées, associez votre chaîne à une configuration de lecture MediaTailor SSAI. Définissez la source de contenu sur l'URL de lecture de votre chaîne.

Conseils

Rubrique	Détail
Placement des pauses publicitaires	Les décalages de pause publicitaire qui ne correspondent pas à la limite d'un segment sont ajustés à la limite la plus proche. Utilisez des segments plus courts (2 à 4 secondes) pour un placement publicitaire plus précis.
Origines des URL signées	MediaTailor met en cache les manifestes des sources de VOD lorsque vous ajoutez la source pour la première fois. Si vous utilisez des URL signées avec des délais d'expiration courts, les URL mises en cache peuvent expirer avant la lecture. Appelez l'UpdateVodSource API pour MediaTailor forcer la récupération du manifeste.
Incompatibilité de rendu	Si la transition d'un programme échoue, vérifiez que toutes les sources VOD de la chaîne ont le même nombre de rendus et utilisent la même échelle ABR.
Planifier à l'avance	Ajoutez des programmes à une chaîne linéaire en cours d'exécution au moins 10 minutes avant leur heure de diffusion. Channel Assembly commence à traiter les programmes dans cette fenêtre.

Pour obtenir des instructions sur la création de votre première chaîne, consultez [the section called “Débuter avec l'assemblage de MediaTailor canaux”](#).

Live-to-live

Une chaîne en direct passe par des flux source en direct sous forme de programmes sur une chaîne programmée. La chaîne passe d'une source en direct à une autre en fonction de votre programme. Utilisez ce flux de travail lorsque l'ensemble de votre contenu est en direct et que vous souhaitez regrouper plusieurs flux en direct sur une seule chaîne linéaire.

Quand l'utiliser

- Vous en avez plusieurs en direct encoders/packageurs et souhaitez passer de l'un à l'autre selon un calendrier.

- Vous créez une chaîne régionale qui bascule entre les flux nationaux et régionaux en direct à différents moments de la journée.
- Vous avez des flux 24/7 en direct (actualités, réseaux sportifs) et vous souhaitez créer une chaîne organisée qui s'appuie sur de multiples sources.

Comment ça marche

1. Configurez votre pipeline d'encodage en direct — Encodez et empaquetez votre contenu en direct à l'aide de AWS Elemental MediaLive AWS Elemental MediaPackage, ou de toute autre origine d'emballage en direct.
2. Créez un emplacement source : enregistrez le serveur d'origine de votre contenu en direct auprès de MediaTailor.
3. Ajouter des sources en direct : enregistrez chaque flux en direct et définissez ses configurations de package et ses groupes de sources.
4. Créer une chaîne — Choisissez le mode de lecture linéaire (requis pour les sources en direct) et configurez les sorties.
5. Programmer des programmes — Ajoutez des programmes source en direct à la programmation de votre chaîne. Spécifiez la durée de chaque émission en direct.
6. Démarrez votre chaîne et configurez un CDN — Démarrez la chaîne. Vous pouvez éventuellement ajouter un CDN devant celui-ci pour des économies d'échelle et de coûts.

Mode de lecture

Live-to-live les chaînes nécessitent le mode de lecture linéaire (niveau standard). Le mode boucle ne prend pas en charge les sources en direct.

Exigences relatives au contenu

Outre les exigences générales énumérées dans [the section called “Utilisation des emplacements sources”](#) :

- Sources en direct HLS : vous devez fournir des #EXT-X-PROGRAM-DATE-TIME balises pour le premier segment de la fenêtre du manifeste et pour chaque discontinuité.
- Marqueurs publicitaires HLS : vous devez configurer les marqueurs publicitaires comme DATERANGE.

- **Fenêtre manifeste source** : utilisez une durée de fenêtre manifeste supérieure d'au moins 30 secondes à celle de la fenêtre manifeste de votre canal Channel Assembly.
- **Durée cible** : faites correspondre la durée cible à la durée des sources existantes de la chaîne.
- **Listes de lecture pour enfants** — Faites correspondre le nombre de listes de lecture pour enfants à celui des autres sources de la chaîne.

Configuration de la source en direct avec MediaPackage

Si vous utilisez AWS Elemental MediaPackage comme origine de votre emballage, nous vous recommandons la configuration suivante :

Paramètres HLS :

Paramètre	Value	Nécessité	Remarques
Type de point de terminaison	Apple HLS ou CMAF	Obligatoire	À utiliser Apple HLS pour les segments TS, CMAF pour les segments MP4
ProgramDateTimeIntervalSeconds	1	Obligatoire	Évite les problèmes de lecture lors des discontinuités.
PlaylistWindowSeconds	30 secondes de plus que la fenêtre du manifeste CA	Obligatoire	
AdMarkers	DATERANGE	Obligatoire	Obligatoire lors du passage par des marqueurs publicitaires.
IncludeIframeOnlyStream	Désactivé	Recommandée	

Réglages DASH :

Paramètre	Value	Nécessité	Remarques
ManifestLayout	FULL	Recommandée	
SegmentTemplateFormat	NUMBER_WITH_TIMELINE ou TIME_WITH_TIMELINE	Recommandée	NUMBER_WITH_DURATION n'est pas pris en charge.
ManifestWindowSeconds	30 secondes de plus que la fenêtre du manifeste CA	Obligatoire	
PeriodTriggers	ADS	Obligatoire	Obligatoire lors du passage par des marqueurs publicitaires.

Pauses publicitaires

Pour les sources en direct, vous ne définissez pas de pauses publicitaires dans le calendrier de Channel Assembly. Au lieu de cela, les pauses publicitaires dans les manifestes des sources en direct sont transmises automatiquement :

- HLS — Les pauses publicitaires utilisant la EXT-X-DATERANGE balise sont transmises.
- DASH — L'utilisation EventStream des pauses publicitaires est transmise.

Pour monétiser les pauses publicitaires transmises, associez votre chaîne à une MediaTailor configuration de lecture SSAI.

Conseils

Rubrique	Détail
Durée du programme	Vous devez spécifier une durée pour chaque programme source en direct. À l'expiration de la durée, Channel Assembly passe au programme suivant du calendrier.

Rubrique	Détail
Ardoise de remplissage	S'il y a des écarts entre les programmes dans l'horaire, jouez MediaTailor sur une liste complète. Si la durée de l'ardoise est inférieure à la durée de l'écart, MediaTailor boucle l'ardoise. Configurez la liste de remplissage lorsque vous créez votre chaîne.
Chaînes régionales	Utilisez des sources en direct pour créer des chaînes régionales. Exécutez-en un encodeur/packageur pour le contenu national, puis exécutez des encodeurs régionaux lorsque ces régions sont en ligne. Créez une chaîne distincte par région, chacune ayant sa propre programmation.
Calendrier de transition	MediaTailor lance la lecture d'un programme sur une limite de segment commune. Pour les transitions absolues, attendez-vous à un segment d'écart entre l'heure planifiée et la transition réelle.
Coûts d'encodage	Réduisez les coûts en utilisant des encodeurs uniquement lorsque le contenu est en ligne. Configurez une liste de personnes à remplir sur votre chaîne pour combler les écarts entre les événements en direct.

VOD + Live-to-live (Mixte)

Une chaîne mixte combine à la fois des sources VOD et des sources en direct dans un seul programme. Il s'agit du type de flux de travail le plus flexible. Utilisez-le lorsque votre chaîne propose principalement du contenu VOD avec des événements en direct programmés entrecoupés, ou lorsque vous souhaitez combler le fossé entre des événements en direct et du contenu préenregistré.

Quand l'utiliser

- Votre chaîne est principalement composée de contenu VOD, mais vous avez programmé des événements en direct (actualités nocturnes, sports hebdomadaires, émissions spéciales).
- Vous souhaitez passer du contenu VOD à la diffusion en direct tout au long de la journée.
- Vous êtes en train de créer une chaîne qui utilise du contenu préenregistré entre les fenêtres d'événements en direct.

Comment ça marche

1. Configurez votre pipeline de contenu : encodez et regroupez à la fois vos ressources VOD et votre contenu en direct. Hébergez-les sur des serveurs d'origine.
2. Créez des localisations sources : enregistrez vos serveurs d'origine auprès de MediaTailor. Vous pouvez utiliser des emplacements sources distincts pour la VOD et le contenu en direct, ou un emplacement source unique s'ils partagent la même origine.
3. Ajoutez des sources de VOD et des sources en direct — Enregistrez vos ressources VOD et vos flux en direct, chacun avec ses configurations de package et ses groupes de sources.
4. Créer une chaîne — Choisissez le mode de lecture linéaire (obligatoire lorsque vous utilisez des sources en direct) et configurez les sorties.
5. Programmez des programmes — Ajoutez un mélange de VOD et de programmes source en direct à la programmation de votre chaîne. Utilisez des transitions absolues pour les programmes en direct qui doivent démarrer à des heures précises. Utilisez des transitions relatives pour les programmes de VOD qui comblent les lacunes.
6. Configurer les pauses publicitaires — Ajoutez des pauses publicitaires aux programmes de VOD. Pour les émissions en direct, les pauses publicitaires sont transmises à partir du manifeste source.
7. Démarrez votre chaîne et configurez un CDN — Démarrez la chaîne. Vous pouvez éventuellement ajouter un CDN devant celui-ci pour des économies d'échelle et de coûts.

Mode de lecture

Les canaux mixtes nécessitent le mode de lecture linéaire (niveau standard). Le mode boucle ne prend pas en charge les sources en direct.

Exigences relatives au contenu

Les chaînes mixtes doivent satisfaire aux exigences en matière de contenu à la fois pour la VOD et pour les sources en direct :

- Toutes les sources (VOD et live) d'un groupe de sources doivent avoir le même nombre de rendus.
- Les vidéos à la demande et les sources en direct doivent avoir des durées de segment cible identiques.
- Les sources en direct doivent répondre aux exigences relatives aux sources en direct, y compris les EXT-X-PROGRAM-DATE-TIME balises pour le HLS.

Schémas de planification

Les chaînes mixtes prennent en charge deux types de transition pour les programmes :

- **Relatif** : le programme est diffusé avant ou après un autre programme de la programmation. Disponible uniquement pour les programmes sources de VOD.
- **Absolu** — Le programme est diffusé à une heure précise sur l'horloge murale. Disponible à la fois pour les programmes VOD et les programmes source en direct en mode linéaire.

Modèles de planification courants :

Modèle	Description	Exemple
Remplissage de VOD avec fenêtres en direct programmées	Les programmes de VOD complètent le calendrier, avec des programmes en direct insérés à des heures précises.	Une chaîne de cinéma qui passe à un événement sportif en direct à 20h00.
Primaire en direct avec interstitiels VOD	Une retransmission en direct est diffusée la plupart du temps, avec des programmes de VOD pendant les pauses ou en dehors des heures de travail.	Chaîne d'information qui diffuse des segments préenregistrés entre les diffusions en direct.
Time-of-day commutation	Différentes sources (en direct ou VOD) sont diffusées à différents moments de la journée.	Diffusion nationale aux heures de grande écoute, diffusion régionale en direct le matin, contenu VOD pendant la nuit.

Pauses publicitaires

Le comportement des pauses publicitaires dépend du type de source de chaque programme :

- **Programmes de VOD** : vous définissez des pauses publicitaires dans le planning avec un décalage en millisecondes. Connectez-vous à une configuration SSAI pour des publicités personnalisées.

- Programmes en direct — Les pauses publicitaires sont automatiquement transmises à partir du manifeste source en direct. HLS utilise des EXT-X-DATERANGE balises ; DASH utilise EventStream.

Conseils

Rubrique	Détail
Transition de la VOD à la diffusion en direct	Lors de la transition d'une émission de VOD à une émission en direct à une heure absolue, MediaTailor tronque le programme de VOD à la limite de segment commun la plus proche de l'heure de début de l'émission en direct.
Transition du live à la VOD	Lorsque la durée d'une émission en direct expire MediaTailor , passe à l'émission suivante de la programmation. Si le programme suivant est de la VOD, la lecture est fluide tant que les exigences en matière de contenu sont respectées.
Ardoise de remplissage	Configurez Filler Slate sur votre chaîne. MediaTailor joue sur ardoise pendant les intervalles entre les programmes. Si la source en direct devient indisponible en cours de programme, Channel Assembly continue de diffuser les derniers segments connus jusqu'à l'expiration de la durée du programme.
Monétisation des publicités mixtes	Vous pouvez utiliser une seule configuration SSAI pour l'ensemble du canal. MediaTailor gère à la fois les pauses publicitaires programmées (programmes VOD) et les pauses publicitaires transmises (programmes en direct) dans le même flux.
Flexibilité des horaires	Ajoutez les programmes de VOD au moins 10 minutes avant l'heure de diffusion. Pour les émissions en direct utilisant des transitions absolues, programmez-les au moins 15 minutes à l'avance sur une chaîne en cours.

Choisir le bon flux de travail

Utilisez le guide de décision suivant pour choisir le type de flux de travail adapté à votre cas d'utilisation :

- Si vous n'avez que du contenu VOD et que vous n'avez pas besoin d'une programmation basée sur le temps → Utilisez le mode VOD-to-live de lecture en boucle.
- Si vous n'avez que du contenu VOD et que vous avez besoin d'une programmation en fonction du temps → Utilisez le mode VOD-to-live de lecture linéaire.
- Si vous n'avez que du contenu en direct → Utilisez Live-to-live.
- Si vous avez à la fois de la VOD et du contenu en direct → Utilisez VOD + Live-to-live (Mixte).

Comparaison des fonctionnalités

Fonctionnalité	VOD-to-live (Boucle)	VOD-to-live (Linéaire)	Live-to-live	Mixte
Sources de VOD	Oui	Oui	Non	Oui
Sources en direct	Non	Non	Oui	Oui
Transitions absolues	Non	Oui	Oui	Oui
Transitions relatives	Oui	Oui	Non	Oui (VOD uniquement)
Pauses publicitaires programmées	Oui	Oui	Non	Oui (VOD uniquement)
Pass-through pauses publicitaires	Non	Non	Oui	Oui (en direct uniquement)
Ardoise de remplissage	Non	Oui	Oui	Oui
24/7 en boucle	Oui	Non	Non	Non
Time-based planification	Non	Oui	Oui	Oui

Fonctionnalité	VOD-to-live (Boucle)	VOD-to-live (Linéaire)	Live-to-live	Mixte
Niveau de chaîne	Base	Standard	Standard	Standard

Étapes suivantes

- [the section called “Débuter avec l'assemblage de MediaTailor canaux”](#)— Step-by-step tutoriel pour créer votre première chaîne
- [the section called “Utilisation des emplacements sources”](#)— Procédures détaillées pour la gestion des emplacements sources et du contenu
- [the section called “Utilisation des chaînes”](#)— Options de configuration des canaux et modes de lecture
- [the section called “Ajouter un programme”](#)— Programmation de programmes avec VOD et sources en direct
- [the section called “ Insérer des publicités et des pauses publicitaires”](#)— Monétisation avec SSAI
- [the section called “Assemblage de canaux avec CDN”](#)— Intégration du CDN pour des économies d'échelle et de coûts
- [the section called “Décalage temporel de lecture d'une chaîne”](#)— Activez les fonctionnalités de rattrapage et de redémarrage

Utilisation des emplacements sources

Un emplacement source représente le serveur d'origine sur lequel votre contenu source est stocké. Un emplacement source peut être Amazon S3, un serveur Web standard, un réseau de diffusion de contenu (CDN) tel qu'Amazon CloudFront ou une origine d'emballage telle que AWS Elemental MediaPackage. MediaTailor récupère vos manifestes de contenu depuis l'emplacement source et les utilise pour assembler le flux linéaire de votre chaîne.

Cette rubrique explique comment utiliser la AWS Elemental MediaTailor console pour créer et supprimer des emplacements de source, et comment travailler avec des sources de VOD.

Rubriques

- [Création d'un emplacement source](#)

- [Configuration de l'authentification pour votre emplacement source](#)
- [Utilisation de sources de VOD](#)
- [Utilisation de sources en direct](#)
- [Utilisation des configurations de packages](#)
- [Mise en cache du manifeste](#)

Création d'un emplacement source

La procédure suivante explique comment créer un emplacement source à l'aide de la MediaTailor console. Pour plus d'informations sur la façon de créer des emplacements de source à l'aide de l'API MediaTailor, consultez [CreateSourceLocation](#) la référence de l'AWS Elemental MediaTailor API.

Pour créer un emplacement source

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage des canaux > Emplacements des sources.
3. Dans la barre de navigation, choisissez Create source location.
4. Sous Configuration de l'emplacement de la source, entrez le nom et l'URL de base de votre serveur d'origine :

- Nom : identifiant de votre emplacement source, tel que my-origin.
- URL de base : protocole et URL de base du serveur d'origine sur lequel votre contenu est stocké, tels que https://111111111111.cloudfront.net. L'URL doit être au format HTTP standard, préfixé par http ://ou https ://.

Sélectionnez éventuellement Utiliser Sigv4 pour l'authentification Amazon S3 si votre emplacement source est un compartiment Amazon S3 et si vous souhaitez utiliser AWS Signature Version 4 pour l'authentification d'accès Amazon S3. Pour des informations avancées, consultez [Configuration de l'authentification pour votre emplacement source](#).

5. Sous Configuration de l'accès, configurez éventuellement l'authentification pour votre emplacement source :
 - Type d'accès : sélectionnez le type d'authentification MediaTailor utilisé pour accéder au contenu stocké sur l'origine de l'emplacement source.

- SigV4 pour Amazon S3 : MediaTailor utilise Amazon Signature Version 4 (Sigv4) pour autoriser la demande à votre origine. Pour de plus amples informations, veuillez consulter [the section called “Authentification des demandes adressées à Amazon S3 avec SigV4”](#).
 - Authentification par jeton d'accès Secrets Manager : MediaTailor utilise Secrets Manager et une clé gérée par le AWS KMS client créée, détenue et gérée par vous pour faciliter l'authentification par jeton d'accès entre MediaTailor et votre origine. Pour plus d'informations sur la configuration de l'authentification par jeton d'accès à Secrets Manager, consultez [the section called “Travailler avec AWS Secrets Manager authentification par jeton d'accès”](#).
 - Nom de l'en-tête : spécifiez un nom d'en-tête HTTP. MediaTailor utilise l'en-tête HTTP pour envoyer le jeton d'accès à votre origine dans les demandes de manifeste de contenu. Vous pouvez utiliser n'importe quel nom d'en-tête tant qu'il ne commence pas par x-amz- ou x-amzn-. Si vous effectuez l'intégration avec une [autorisation MediaPackage CDN](#), la valeur de l'en-tête doit être X-MediaPackage-CDNIdentifier.
 - Clé de chaîne secrète : **SecretString** clé que vous avez spécifiée dans le secret de votre Gestionnaire de Secrets. Par exemple, s'il SecretString contient une paire clé/valeur telle que : {"MyHeaderName": "11111111-2222-3333-4444-111122223333"}, MyHeaderName c'est la SecretString clé que vous entrez dans ce champ.
 - ARN secret : ARN du secret qui contient votre jeton d'accès. Pour un guide étape par étape, voir [Étape 2 : Création d'un AWS Secrets Manager secret](#).
6. Sous Configuration du serveur de diffusion de segments, configurez éventuellement un serveur pour diffuser vos segments de contenu :
- Utiliser un serveur de diffusion de segments par défaut : entrez l'URL de base du serveur utilisé pour diffuser vos segments de contenu, tel qu'un CDN. Configurez le nom d'hôte du segment par défaut si vous souhaitez utiliser un serveur différent du serveur de localisation source pour servir les segments de contenu. Par exemple, vous pouvez restreindre l'accès des joueurs aux manifestes d'origine en utilisant une configuration CDN différente pour l'URL HTTP de base (MediaTailor utilisée pour accéder aux manifestes) et l'URL de base du segment par défaut (utilisée par les joueurs pour accéder aux segments de contenu). Si vous ne saisissez aucune valeur, le serveur de localisation source est utilisé MediaTailor par défaut pour la diffusion des segments.
 - Utiliser des serveurs de diffusion de segments nommés : si vous avez configuré un serveur de diffusion de segments par défaut, vous pouvez également configurer des serveurs de diffusion

de segments supplémentaires. Chacun doit avoir un nom unique et une URL de base. L'URL de base peut être une URL HTTP complète ou un chemin relatif, par exemple `/some/path/`. Les noms sont utilisés pour identifier le serveur à utiliser lors de la MediaTailor réception d'une demande de segments de contenu. Si la demande contient l'en-tête `X-MediaTailor-SegmentDeliveryConfigurationName` et que la valeur de l'en-tête correspond à un nom, l'URL de base correspondante sera utilisée pour diffuser le contenu. Si l'en-tête n'est pas inclus dans la demande ou s'il ne correspond à aucun nom, le serveur de distribution de segments par défaut sera utilisé.

7. Choisissez Créer un emplacement source.
8. Pour ajouter d'autres emplacements de source, répétez les étapes 2 à 6.

Configuration de l'authentification pour votre emplacement source

Utilisez la configuration des accès pour configurer l'authentification pour votre emplacement source. Lorsque la configuration des accès est activée, il MediaTailor ne récupère les manifestes source de votre origine que si la demande est autorisée entre MediaTailor et votre origine. La configuration des accès est désactivée par défaut.

MediaTailor prend en charge les types d'authentification suivants :

- SigV4 pour l'authentification Amazon S3
- AWS Secrets Manager jeton d'accès
- SigV4 pour l' MediaPackage authentification de la version 2 (v2)

Ce chapitre explique comment utiliser Sigv4 pour Amazon S3, MediaPackage v2, et les jetons d'AWS Secrets Manager accès pour l'authentification de l'emplacement de la source.

Pour plus d'informations, sélectionnez la rubrique appropriée.

Rubriques

- [Authentification des demandes adressées à Amazon S3 avec SigV4](#)
- [Utilisation de SigV4 pour la MediaPackage version 2](#)
- [Travailler avec AWS Secrets Manager authentification par jeton d'accès](#)

Authentification des demandes adressées à Amazon S3 avec SigV4

Signature Version 4 (Sigv4) pour Amazon S3 est un protocole de signature utilisé pour authentifier les demandes adressées à Amazon S3 via HTTPS. Lorsque vous utilisez SigV4 pour Amazon S3, MediaTailor inclut un en-tête d'autorisation signé dans la demande HTTPS envoyée au compartiment Amazon S3 utilisé comme origine. Si l'en-tête d'autorisation signé est valide, votre origine répond à la demande. Si elle n'est pas valide, la demande échoue.

Pour obtenir des informations générales sur SigV4 pour AWS Key Management Service, consultez la rubrique [Authentication Requests \(AWS Signature Version 4\)](#) dans le manuel de référence de l'API Amazon S3.

Note

MediaTailor signe toujours les demandes adressées à ces origines avec SigV4.

Exigences

Si vous activez SigV4 pour l'authentification Amazon S3 pour votre emplacement source, vous devez répondre aux exigences suivantes :

- Vous devez autoriser MediaTailor l'accès à votre compartiment Amazon S3 en accordant à `mediatailor.amazonaws.com` un accès principal dans IAM. Pour plus d'informations sur la configuration de l'accès dans IAM, consultez la section [Gestion des accès](#) dans le Guide de Gestion des identités et des accès AWS l'utilisateur.
- Le responsable du service `mediatailor.amazonaws.com` doit être autorisé à lire toutes les playlists multivariantes référencées par les configurations du package source VOD.
- L'appelant de l'API doit disposer des autorisations `s3 : GetObject` IAM pour lire toutes les playlists multivariantes référencées par les configurations de votre package source MediaTailor VOD.
- L'URL de base de votre localisation MediaTailor source doit respecter le format d'URL de demande de type hébergé virtuel Amazon S3. Par exemple, `https://bucket-name.s3.Region.amazonaws.com/key-name`. Pour plus d'informations sur l'accès de type virtuel hébergé par Amazon S3, consultez [Virtual Hosted-Style](#) Requests.

Utilisation de SigV4 pour la MediaPackage version 2

Signature Version 4 (SigV4) pour MediaPackage v2 est un protocole de signature utilisé pour authentifier les demandes adressées à la MediaPackage v2 via HTTP. Lorsque vous utilisez SigV4 pour la MediaPackage version v2, MediaTailor inclut un en-tête d'autorisation signé dans la requête HTTP envoyée au point de terminaison MediaPackage v2 utilisé comme origine. Si l'en-tête d'autorisation signé est valide, votre origine répond à la demande. Si elle n'est pas valide, la demande échoue.

Pour des informations générales sur SigV4 pour MediaPackage v2, consultez la rubrique [Authentification des demandes \(AWS signature version 4\)](#) dans le guide de référence de l'API MediaPackage v2.

Exigences

Si vous activez SigV4 pour l'authentification MediaPackage v2 pour votre emplacement source, vous devez répondre aux exigences suivantes :

- Vous devez autoriser l'accès MediaTailor à votre point de terminaison MediaPackage v2 en accordant l'accès principal à `mediatailor.amazonaws.com` dans le cadre d'une politique d'accès à Origin sur le point de terminaison.
- L'URL de base de votre localisation MediaTailor source doit être un point de terminaison MediaPackage v2.
- L'appelant de l'API doit disposer des autorisations `mediapackagev2 : GetObject` IAM pour lire toutes les playlists multivariantes référencées par les configurations du package source. MediaTailor

Travailler avec AWS Secrets Manager authentification par jeton d'accès

MediaTailor prend en charge l'authentification par jeton d'accès à Secrets Manager. L'authentification par jeton d' AWS Secrets Manager accès MediaTailor utilise une clé AWS Key Management Service (AWS KMS) gérée par le client et un AWS Secrets Manager secret que vous créez, détenez et gérez pour authentifier les demandes auprès de votre origine.

Dans cette section, nous expliquons le fonctionnement de l'authentification par jeton d'accès à Secrets Manager et fournissons des informations détaillées sur la manière de configurer l'authentification par jeton d'accès à Secrets Manager. Vous pouvez utiliser l'authentification par jeton d'accès à Secrets Manager dans Console de gestion AWS ou par programmation avec AWS les API.

Rubriques

- [Configuration AWS Secrets Manager authentification par jeton d'accès](#)
- [Intégration aux MediaPackage points de terminaison qui utilisent l'autorisation CDN](#)
- [Comment fonctionne l'authentification par jeton d'accès à MediaTailor Secrets Manager](#)

Configuration AWS Secrets Manager authentification par jeton d'accès

Lorsque vous souhaitez utiliser l'authentification par jeton d' AWS Secrets Manager accès, vous devez suivre les étapes suivantes :

1. Vous [créez une clé gérée par le AWS Key Management Service client](#).
2. Tu [crées un AWS Secrets Manager secret](#). Le secret contient votre jeton d'accès, qui est stocké dans Secrets Manager sous forme de valeur secrète cryptée. MediaTailor utilise la clé gérée par le AWS KMS client pour déchiffrer la valeur secrète.
3. Vous configurez un emplacement AWS Elemental MediaTailor source pour utiliser l'authentification par jeton d'accès à Secrets Manager.

La section suivante fournit des instructions détaillées sur la façon de configurer l'authentification par jeton d' AWS Secrets Manager accès.

Rubriques

- [Étape 1 : Création d'un AWS KMS clé symétrique gérée par le client](#)
- [Étape 2 : Création d'un AWS Secrets Manager secret](#)
- [Étape 3 : Configuration d'un emplacement MediaTailor source avec authentification par jeton d'accès](#)

Étape 1 : Création d'un AWS KMS clé symétrique gérée par le client

Vous l'utilisez AWS Secrets Manager pour stocker votre jeton d'accès sous la forme d'un code secret. SecretString SecretString est chiffré à l'aide d'une cléAWS KMS symétrique gérée par le client que vous créez, détenez et gérez. MediaTailor utilise la clé symétrique gérée par le client pour faciliter l'accès au secret avec une autorisation, et pour chiffrer et déchiffrer la valeur du secret.

Les clés gérées par le client vous permettent d'effectuer des tâches telles que les suivantes :

- Établissement et gestion des stratégies de clé

- Établissement et gestion des politiques IAM et des octrois
- Activation et désactivation des stratégies de clé
- Matériau clé cryptographique rotatif
- Ajout de balises

Pour plus d'informations sur la manière dont Secrets Manager protège les secrets, consultez la rubrique [Comment les AWS Secrets Manager utiliser AWS KMS](#) dans le Guide du AWS Key Management Service développeur. AWS KMS

Pour plus d'informations sur les clés gérées par le client, consultez [Clés gérées par le client](#) dans le Guide du développeur AWS Key Management Service .

 Note

AWS KMS des frais s'appliquent pour l'utilisation d'une clé gérée par le client. Pour plus d'informations sur les tarifs, consultez la page de [AWS Key Management Service tarification](#).

Vous pouvez créer une clé AWS KMS symétrique gérée par le client à l'aide des API Console de gestion AWS ou par programmation à l'aide de celles-ci. AWS KMS

Pour créer une clé symétrique gérée par le client

Suivez les étapes de [création d'une clé symétrique gérée par le client](#) dans le guide du AWS Key Management Service développeur.

Notez la clé Amazon Resource Name (ARN) ; vous en aurez besoin [Étape 2 : Création d'un AWS Secrets Manager secret](#).

Contexte de chiffrement

Un contexte de chiffrement est un ensemble facultatif de paires clé-valeur qui contient des informations contextuelles supplémentaires sur les données.

Secrets Manager inclut un [contexte de chiffrement](#) lors du chiffrement et du déchiffrement du. `SecretString` Le contexte de chiffrement inclut l'ARN secret, qui limite le chiffrement à ce secret spécifique. Comme mesure de sécurité supplémentaire, MediaTailor crée une AWS KMS subvention en votre nom. MediaTailor applique une [GrantConstraints](#) opération qui nous permet uniquement

de déchiffrer l'ARN secret `SecretString` associé au contenu du contexte de chiffrement Secrets Manager.

Pour plus d'informations sur la manière dont Secrets Manager utilise le contexte de [chiffrement](#), [consultez la rubrique Contexte](#) de chiffrement du Guide du AWS Key Management Service développeur.

Définition de la politique clé

Les stratégies de clés contrôlent l'accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une stratégie 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 utiliser la politique de clé par défaut. Pour plus d'informations, consultez [Authentification et contrôle d'accès pour AWS KMS](#) dans le Guide du développeur AWS Key Management Service .

Pour utiliser votre clé gérée par le client avec vos ressources de localisation MediaTailor source, vous devez autoriser le principal IAM qui appelle [CreateSourceLocation](#) ou utilise [UpdateSourceLocation](#) les opérations d'API suivantes :

- `kms:CreateGrant`— Ajoute une autorisation à une clé gérée par le client. MediaTailor crée une autorisation sur votre clé gérée par le client qui lui permet d'utiliser la clé pour créer ou mettre à jour un emplacement source configuré avec une authentification par jeton d'accès. Pour plus d'informations sur l'utilisation de [Grants dans AWS KMS](#), consultez le guide du AWS Key Management Service développeur.

Cela permet MediaTailor d'effectuer les opérations suivantes :

- Appelez `Decrypt` pour qu'il puisse récupérer avec succès le secret de votre Gestionnaire de Secrets Manager lors de l'appel [GetSecretValue](#).
- Appelez `RetireGrant` pour annuler l'autorisation lorsque l'emplacement source est supprimé ou lorsque l'accès au secret a été révoqué.

Voici un exemple de déclaration de politique que vous pouvez ajouter pour MediaTailor :

```
{
  "Sid": "Enable MediaTailor Channel Assembly access token usage for the
MediaTailorManagement IAM role",
  "Effect": "Allow",
  "Principal": {
```



```

    "AWS": "arn:aws:iam::account number:role/MediaTailorManagement"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": "mediatailor.region.amazonaws.com"
    }
  }
}

```

Pour plus d'informations sur la définition des autorisations dans une politique et la résolution des problèmes d'accès par clé, consultez la section [Subventions AWS KMS dans](#) le guide du AWS Key Management Service développeur.

Étape 2 : Création d'un AWS Secrets Manager secret

Utilisez Secrets Manager pour stocker votre jeton d'accès sous la forme d'un `SecretString` jeton chiffré par une clé gérée par le AWS KMS client. MediaTailor utilise la clé pour déchiffrer le `SecretString`. Pour plus d'informations sur la manière dont Secrets Manager protège les secrets, consultez la rubrique [Comment les AWS Secrets Manager utiliser AWS KMS](#) dans le Guide du AWS Key Management Service développeur. AWS KMS

Si vous utilisez AWS Elemental MediaPackage comme emplacement source l'origine et que vous souhaitez utiliser l'authentification par jeton d'accès MediaTailor Secrets Manager, suivez la procédure [the section called “Intégration aux MediaPackage points de terminaison qui utilisent l'autorisation CDN”](#).

Vous pouvez créer un secret Secrets Manager à l'aide des API Secrets Manager Console de gestion AWS ou par programmation à l'aide des API Secrets Manager.

Pour créer un secret

Suivez les étapes de [création et de gestion des secrets décrites AWS Secrets Manager](#) dans le guide de AWS Secrets Manager l'utilisateur.

Tenez compte des considérations suivantes lors de la création de votre secret :

- [KmsKeyId](#) doit s'agir de l'[ARN de la clé](#) gérée par le client que vous avez créée à l'étape 1.
- Vous devez fournir un [SecretString](#). `SecretString` doit s'agir d'un objet JSON valide qui inclut une clé et une valeur contenant le jeton d'accès. Par exemple, {» MyAccessTokenIdentifier « ? 112233445566"}. La valeur doit comporter entre 8 et 128 caractères.

Lorsque vous configurez votre emplacement source avec l'authentification par jeton d'accès, vous spécifiez la `SecretString` clé. MediaTailor utilise la clé pour rechercher et récupérer le jeton d'accès stocké dans le `SecretString`.

Notez l'ARN secret et la `SecretString` clé. Vous les utiliserez lorsque vous configurerez votre emplacement source pour utiliser l'authentification par jeton d'accès.

Joindre une politique secrète basée sur les ressources

Pour autoriser l' MediaTailor accès à la valeur secrète, vous devez associer une politique basée sur les ressources à la clé secrète. Pour plus d'informations, consultez la section [Attacher une politique d'autorisations à un secret de Secrets Manager](#) dans le Guide de AWS Secrets Manager l'utilisateur.

Voici un exemple de déclaration de politique que vous pouvez ajouter pour MediaTailor :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "mediatailor.amazonaws.com"
      },
      "Action": "secretsmanager:GetSecretValue",
      "Resource": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"
    }
  ]
}
```

Étape 3 : Configuration d'un emplacement MediaTailor source avec authentification par jeton d'accès

Vous pouvez configurer l'authentification par jeton d'accès à Secrets Manager à l'aide des API Console de gestion AWS ou par programmation. MediaTailor

Pour configurer un emplacement source avec l'authentification par jeton d'accès à Secrets Manager

Suivez les étapes décrites [Access configuration](#) dans le guide de AWS Elemental MediaTailor l'utilisateur.

Intégration aux MediaPackage points de terminaison qui utilisent l'autorisation CDN

Si vous utilisez AWS Elemental MediaPackage comme origine de localisation source, vous MediaTailor pouvez intégrer les MediaPackage points de terminaison qui utilisent l'autorisation CDN.

Pour intégrer un MediaPackage point de terminaison qui utilise l'autorisation CDN, suivez la procédure suivante.

Pour intégrer à MediaPackage

1. Si ce n'est pas déjà fait, suivez les étapes décrites dans la section [Configuration de l'autorisation CDN](#) dans le guide de l'AWS Elemental MediaPackage utilisateur.
2. Exécutez la procédure dans [the section called “Étape 1 : Création d'un AWS KMS clé symétrique gérée par le client”](#).
3. Modifiez le secret que vous avez créé lors de la configuration de l'autorisation du MediaPackage CDN. Modifiez le secret avec les valeurs suivantes :
 - Mettez à jour le KmsKeyId avec la clé ARN gérée par le client dans laquelle vous l'avez créée [the section called “Étape 1 : Création d'un AWS KMS clé symétrique gérée par le client”](#).
 - (Facultatif) Pour le SecretString, vous pouvez soit faire pivoter l'UUID vers une nouvelle valeur, soit utiliser le secret chiffré existant tant qu'il s'agit d'une paire clé/valeur au format JSON standard, tel que. {"MediaPackageCDNIdentifier": "112233445566778899"}
4. Suivez les étapes de [the section called “Joindre une politique secrète basée sur les ressources”](#).
5. Suivez les étapes de [the section called “Étape 3 : Configuration d'un emplacement MediaTailor source avec authentification par jeton d'accès”](#).

Comment fonctionne l'authentification par jeton d'accès à MediaTailor Secrets Manager

Une fois que vous avez créé ou mis à jour un emplacement source pour utiliser l'authentification par jeton d'accès, MediaTailor incluez le jeton d'accès dans un en-tête HTTP lorsque vous demandez des manifestes de contenu source depuis votre origine.

Voici un aperçu de la façon dont l'authentification par jeton d'accès Secrets Manager est MediaTailor utilisée pour l'authentification de l'origine de l'emplacement de la source :

1. Lorsque vous créez ou mettez à jour un emplacement MediaTailor source qui utilise l'authentification par jeton d'accès, MediaTailor envoie une [DescribeSecret](#) demande à Secrets Manager pour déterminer la AWS KMS clé associée au secret. Vous incluez l'ARN secret dans la configuration d'accès à votre emplacement source.
2. MediaTailor crée une [autorisation](#) pour la clé gérée par le client, afin que celui-ci MediaTailor puisse utiliser la clé pour accéder au jeton d'accès stocké dans le SecretString. Le nom de la subvention sera `MediaTailor-SourceLocation-your Compte AWS ID-source location name`.

Vous pouvez révoquer l'accès à l'autorisation ou supprimer MediaTailor l'accès à la clé gérée par le client à tout moment. Pour plus d'informations, consultez [RevokeGrant](#) dans la Référence d'API AWS Key Management Service .

3. Lorsqu'une source VOD est créée, mise à jour ou utilisée dans un programme, envoie MediaTailor des requêtes HTTP aux emplacements source pour récupérer les manifestes de contenu source associés aux sources VOD dans l'emplacement source. Si la source VOD est associée à un emplacement source pour lequel un jeton d'accès est configuré, les demandes incluent le jeton d'accès en tant que valeur d'en-tête HTTP.

Utilisation de sources de VOD

Une source VOD représente un contenu unique, tel qu'une vidéo ou un épisode d'un podcast, que vous ajoutez à votre emplacement source. Vous ajoutez une ou plusieurs sources de VOD à l'emplacement de votre source, puis vous associez chaque source VOD à un programme après avoir créé votre chaîne.

Chaque source VOD doit avoir au moins une configuration de package. Une configuration de package spécifie le format du package, l'emplacement du manifeste et le groupe de sources pour votre source VOD. Lorsque vous créez votre chaîne, vous utilisez les groupes de sources de la configuration du package pour créer les sorties correspondantes sur votre chaîne. Par exemple, si votre source est emballée dans deux formats différents (HLS et DASH), vous devez créer deux configurations de package, l'une pour DASH et l'autre pour HLS. Ensuite, vous devez créer deux sorties de canal, une pour chaque configuration de package. Chaque sortie de canal fournit un point de terminaison utilisé pour les demandes de lecture. Ainsi, en utilisant l'exemple précédent, le canal fournirait un point de terminaison pour les demandes de lecture HLS et un point de terminaison pour les demandes de lecture DASH.

Si vous souhaitez que les décalages des marqueurs publicitaires de votre manifeste soient détectés automatiquement, chaque marqueur publicitaire doit apparaître au même décalage dans toutes les configurations de package et avoir une durée de zéro. Pour le HLS, MediaTailor détectera DATERANGE et balisera EXT-X-CUE-OUT. Pour DASH, le HLS détectera la première balise Event dans chaque EventStream balise.

Dans l'exemple suivant, une opportunité de rupture publicitaire sera détectée avec un décalage de 12 000 ms en raison de la durée de la DATERANGE balise de 0,0. La première DATERANGE balise décalée de 0 ms ne sera pas détectée car elle a une durée de 10,0.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-DATERANGE:ID="1001",START-DATE="2021-09-16T23:51:05.249Z",DURATION=10.0,SCTE35-
OUT=0xFC302500000003289800FFF01405000003E97FEFFE1D381BD8FE000DBBA00001010100000FD2B275
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_0.ts
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_1.ts
#EXT-X-DATERANGE:ID="1001",START-DATE="2021-09-16T23:51:05.249Z",DURATION=0.0,SCTE35-
OUT=0xFC302500000003289800FFF01405000003E97FEFFE1D381BD8FE000DBBA00001010100000FD2B275
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_2.ts
```

Dans l'exemple suivant, une opportunité de rupture publicitaire sera détectée avec un décalage de 0 ms, car le EXT-X-CUE-OUT tag a une durée de 0 et est immédiatement suivi d'un EXT-X-CUE-IN tag. La seconde EXT-X-CUE-IN paireEXT-X-CUE-OUT/ne sera pas détectée car elle a une durée de 10.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-CUE-OUT:0
#EXT-X-CUE-IN
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_0.ts
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_1.ts
#EXT-X-CUE-OUT:10
```

```
...
#EXT-X-CUE-IN
#EXTINF:6.000,
../.../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_2.ts
```

Dans l'exemple suivant, une opportunité de rupture publicitaire sera détectée avec un décalage de 0 ms car le premier événement de la série EventStream se produit au cours de la période commençant à PT0.000S. Le deuxième Event d'entre eux ne EventStream sera pas détecté.

```
<Period start="PT0.000S" id="9912561" duration="PT29.433S">
<EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
<Event duration="0">
  <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="1241950593" tier="4095">
    <scte35:SpliceInsert spliceEventId="99" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
      <scte35:Program><scte35:SpliceTime ptsTime="3552273000"/></scte35:Program>
      <scte35:BreakDuration autoReturn="true" duration="2700000"/>
    </scte35:SpliceInsert>
  </scte35:SpliceInfoSection>
</Event>
<Event duration="0">
  <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="1241950593" tier="4095">
    <scte35:SpliceInsert spliceEventId="99" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
      <scte35:Program><scte35:SpliceTime ptsTime="3552273000"/></scte35:Program>
      <scte35:BreakDuration autoReturn="true" duration="2700000"/>
    </scte35:SpliceInsert>
  </scte35:SpliceInfoSection>
</Event>
</EventStream>
...
</Period>
```

Ajouter des sources VOD à votre emplacement source

La procédure suivante explique comment ajouter des sources VOD à votre emplacement source et configurer les configurations de package à l'aide de la MediaTailor console. Pour plus d'informations sur la façon d'ajouter des sources de VOD à l'aide de l' API MediaTailor, consultez [CreateVodSource](#) la référence de l'AWS Elemental MediaTailor API.

 Important

Avant d'ajouter vos sources de VOD, assurez-vous qu'elles répondent aux exigences suivantes :

- Les variantes de source doivent toutes avoir la même longueur, telle que déterminée par le manifeste source.
- Dans la configuration d'un package, chaque source doit avoir le même nombre de flux enfants.

En raison de ces exigences, nous ne prenons pas en charge l'ABR par titre ni l'ABR automatique, car ces méthodes de codage peuvent produire des longueurs de manifeste et des flux enfants variables.

Nous vous recommandons d'utiliser un modèle de codage incluant une longueur de segment minimale afin de garantir que vos sources encodées répondent à ces exigences.

Pour ajouter des sources de VOD à vos emplacements sources

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage des canaux > Emplacements des sources.
3. Dans le volet Emplacements des sources, choisissez l'emplacement source que vous avez créé dans la [Pour créer un emplacement source](#) procédure.
4. Choisissez Ajouter une source VOD.
5. Sous Détails de la source VOD, entrez le nom de votre source VOD :
 - Nom : identifiant de votre source de VOD, tel que my-example-video.
6. Sous Configurations du package > *source-group-name* entrez les informations relatives à la configuration du package :

 Note

Les configurations de package de votre source doivent toutes avoir la même durée, telle que déterminée par le manifeste de la source. De plus, toutes les sources d'une configuration de package doivent avoir le même nombre de flux enfants. Pour répondre à ces exigences, nous vous recommandons d'utiliser un modèle de codage pour vos

actifs. Nous vous recommandons d'utiliser un modèle de codage avec une longueur de segment minimale d'une seconde. MediaTailor ne prend pas en charge le streaming par titre ni le streaming automatique à débit adaptatif (ABR) car ces méthodes de codage ne respectent pas ces exigences.

- **Groupe source** : entrez un nom de groupe source qui décrit cette configuration de package, tel que HLS-4k. Notez ce nom ; vous le référencerez lorsque vous créerez la sortie de votre chaîne. Pour de plus amples informations, veuillez consulter [Utilisation de groupes de sources avec les sorties de votre chaîne](#).
- **Type** : sélectionnez le format du package pour cette configuration. MediaTailor supporte HLS et DASH.
- **Chemin relatif** : chemin relatif entre l'URL HTTP de base de l'emplacement source et le manifeste. Par exemple, /my/path/index.m3u8.

 Note

MediaTailor importe automatiquement tous les sous-titres codés et les flux enfants contenus dans une playlist multivariée parente. Il n'est pas nécessaire de créer des configurations de package distinctes pour chacun de vos rendus sources (DASH) ou flux de variantes (HLS).

Pour plus d'informations sur les configurations de package, consultez [Utilisation des configurations de packages](#).

7. Choisissez Ajouter une source VOD.

Si vous souhaitez ajouter d'autres sources de VOD, répétez les étapes 4 à 7 de la procédure.

Utilisation de sources en direct

Une source en direct représente une diffusion en direct unique, telle qu'un match de football en direct ou une émission d'actualités, que vous ajoutez à votre emplacement source. Après avoir créé votre chaîne, vous ajoutez une ou plusieurs sources en direct à l'emplacement de votre source, puis vous associez chaque source en direct à un programme.

MediaTailor prend en charge les types d'assemblage de canaux linéaires suivants :

- Sources de VOD pour une chaîne contenant VOD-to-live du contenu
- Sources en direct pour une chaîne qui contient du contenu en direct mélangé à du contenu VOD-to-live

Un exemple de VOD-to-live contenu est une chaîne qui assemble une bibliothèque de ressources VOD dans un flux en direct. Un exemple de contenu en direct mélangé à du contenu est une chaîne qui diffuse principalement VOD-to-live du contenu VOD, à l'exception d'un événement d'actualité nocturne ou d'un événement sportif en direct préprogrammé. Un autre exemple de contenu en direct mélangé à du VOD-to-live contenu est une chaîne entièrement en direct dont l'origine varie en fonction de l'heure de la journée.

Vous pouvez utiliser des sources en direct pour configurer une chaîne régionale qui diffuse principalement des émissions nationales, mais inclut également des remplacements de programmation régionale et intègre du contenu VOD. Pour ce faire, vous exécutez une encoder/packageur pour le contenu national, puis vous exécutez des encodeurs régionaux lorsque ces régions sont actives. Vous créez ensuite des chaînes d'assemblage de chaînes régionales, chacune ayant ses propres horaires. De cette façon, les spectateurs peuvent alterner entre eux selon leurs besoins. Cette configuration vous permet de minimiser les encoding/packageur coûts.

Chaque source en direct doit avoir au moins une configuration de package. Une configuration de package spécifie le format du package, l'emplacement du manifeste et le groupe de sources pour votre source en direct. Lorsque vous créez votre chaîne, vous utilisez les groupes de sources de la configuration du package pour créer les sorties correspondantes sur votre chaîne. Par exemple, si votre source est empaquetée dans deux formats différents (HLS et DASH), vous devez créer deux configurations de package, l'une pour DASH et l'autre pour HLS. Ensuite, vous créez deux sorties de canal, une pour chaque configuration de package. Chaque sortie de canal fournit un point de terminaison utilisé pour les demandes de lecture. Dans cet exemple, le canal fournit un point de terminaison pour les demandes de lecture HLS et un point de terminaison pour les demandes de lecture DASH.

Exigences générales relatives à l'utilisation de sources en direct

Lorsque vous utilisez des sources en direct, votre contenu doit respecter les exigences générales suivantes :

- Sources dynamiques HLS : vous devez fournir des #EXT-X-PROGRAM-DATE-TIME balises pour le premier segment de la fenêtre du manifeste et pour chaque discontinuité.
- HLS - Vous devez configurer les marqueurs publicitaires comme DATERANGE.

- Fenêtre de manifeste source : nous vous recommandons d'utiliser une fenêtre de manifeste dont la durée est au moins aussi longue que celle de la fenêtre de manifeste de votre MediaTailor canal Channel Assembly. Il est recommandé d'utiliser une fenêtre de manifeste d'une durée de 30 secondes ou plus que celle de la fenêtre de manifeste sur le canal Channel Assembly.
- Faites en sorte que la durée cible corresponde à la durée des sources existantes.
- Faites en sorte que le nombre de playlists pour enfants corresponde à celui des sources existantes.

Configurations

Si vous utilisez d'autres services multimédias AWS Elemental dans le cadre de votre flux de travail sur les sources en direct, nous vous recommandons de suivre les meilleures pratiques lors de la configuration de votre MediaPackage configuration. Le tableau suivant décrit comment configurer les MediaPackage paramètres en fonction de la norme de streaming que vous utilisez.

MediaPackage configuration pour les sources en direct

Standard	Paramètre	Value	Nécessité	Remarques
HLS	Type de point de terminaison	Apple HLS	Obligatoire sauf si vous utilisez le CMAF	Pour correspondre aux offres d'emploi HLS ts AWS Elemental MediaConvert
HLS	Type de point de terminaison	CMAF	Obligatoire sauf si vous utilisez Apple HLS	Pour correspondre aux offres d'emploi HLS mp4 AWS Elemental MediaConvert
HLS	ProgramDataTimeIntervalSeconds	1	Obligatoire	Vous devez #EXT-X-PROGRAM-DATE-TIME le spécifier sur chaque segment

Standard	Paramètre	Value	Nécessité	Remarques
				afin d'éviter les problèmes de lecture en cas de discontinuité.
HLS	PlaylistWindowSeconds	30 secondes de plus que la fenêtre du manifeste d'assemblage des canaux	Obligatoire	
HLS	AdMarkers	DATERANGE	Obligatoire lors du passage à travers les marqueurs publicitaires	
HLS	IncludeIframeOnlyStream	Désactivé	Recommandée	
DASH	ManifestLayout	FULL	Recommandée	
DASH	SegmentTemplateFormat	NUMBER_WITH_TIMELINE ou TIME_WITH_TIMELINE	Recommandée	NUMBER_WITH_DURATION n'est pas pris en charge.

Standard	Paramètre	Value	Nécessité	Remarques
DASH	ManifestWindowSeconds	30 secondes de plus que la fenêtre du manifeste d'assemblage des canaux	Obligatoire	
DASH	PeriodTriggers	ADS	Obligatoire lors du passage à travers les marqueurs publicitaires	

Ajouter des sources en direct à votre emplacement source

La procédure suivante explique comment utiliser la MediaTailor console pour ajouter des sources dynamiques à votre emplacement source et configurer les configurations de package. Pour plus d'informations sur la façon d'ajouter des sources en direct à l'aide de l' API MediaTailor, consultez [CreateLiveSource](#) la référence de l'AWS Elemental MediaTailor API.

Important

Avant d'ajouter vos sources en direct, assurez-vous que dans la configuration d'un package, chaque source possède le même nombre de flux enfants.

Pour ajouter des sources en direct à vos emplacements sources

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage des canaux > Emplacements des sources.
3. Dans le volet Emplacements des sources, choisissez l'emplacement source que vous avez créé dans la [Pour créer un emplacement source](#) procédure.
4. Dans l'onglet Sources dynamiques, choisissez Ajouter une source en direct.
5. Sous Détails de la source en direct, entrez le nom de votre source en direct :

- Nom : identifiant de votre source en direct, tel que my-example-video.
6. Sous Configurations du package > *source-group-name* entrez les informations relatives à la configuration du package :

 Note

Dans la configuration d'un package, toutes les sources VOD et les sources en direct doivent avoir le même nombre de flux enfants. Nous vous recommandons de configurer vos flux sources de la même manière.

- Groupe source : entrez un nom de groupe source qui décrit cette configuration de package, tel que HLS-4k. Notez ce nom ; vous le référencerez lorsque vous créerez la sortie de votre chaîne. Pour de plus amples informations, veuillez consulter [Utilisation de groupes de sources avec les sorties de votre chaîne](#).
- Type : sélectionnez le format du package pour cette configuration. MediaTailor supporte HLS et DASH.
- Chemin relatif : chemin relatif entre l'URL HTTP de base de l'emplacement source et le manifeste. Par exemple, /my/path/index.m3u8.

 Note

MediaTailor importe automatiquement tous les sous-titres codés et les flux enfants contenus dans une playlist multivariée parente. Il n'est pas nécessaire de créer des configurations de package distinctes pour chacun de vos rendus sources (DASH) ou flux de variantes (HLS).

Pour plus d'informations sur les configurations de package, consultez [Utilisation des configurations de packages](#).

7. Choisissez Ajouter une source en direct.

Si vous souhaitez ajouter d'autres sources en direct, répétez les étapes 4 à 6 de la procédure.

Utilisation des configurations de packages

Une configuration de package est une représentation de la source qui contient les différentes caractéristiques d'emballage requises pour la lecture sur différents appareils. Par exemple, vous pouvez avoir une source qui possède trois formats packagés : HLS avec DRM, DASH avec adressage chronologique des segments et HLS avec segments CMAF.

L'assemblage des canaux ne reconditionne pas vos sources. Si vous souhaitez inclure plusieurs formats de package pour une source donnée, vous devez rendre chaque format de package disponible à l'emplacement de la source et spécifier le chemin d'accès à chaque format de package.

Chaque objet de configuration de package doit inclure les éléments suivants :

- Chemin relatif : chemin complet vers le format empaqueté de la source, par rapport à l'emplacement de la source. Par exemple, `/my/path/index.m3u8`.
- Groupe de sources : nom du groupe de sources utilisé pour associer les configurations de package à la sortie d'un canal.
- Type : HLS ou DASH.

Après avoir créé un canal, vous devez également déclarer chaque groupe de sources que vous souhaitez utiliser pour la sortie du canal.

Mise en cache du manifeste

MediaTailor met en cache périodiquement et de manière opportuniste les playlists sources afin d'améliorer les performances et la fiabilité de l'assemblage des chaînes. Parfois, la version mise en cache devient obsolète par rapport à la version d'origine à votre emplacement source. MediaTailor Pour forcer l'actualisation de la version mise en cache de la source, appelez [UpdateVodSource](#). Par exemple, utilisez cet appel lorsque les chemins intégrés changent dans votre source. Assurez-vous de toujours conserver une version à jour de la source disponible sur votre site d'origine, même si vous recevez peu de demandes de la part de la source MediaTailor.

Utilisation des chaînes

Un canal assemble vos manifestes sources en un flux linéaire. Chaque canal contient une ou plusieurs sorties correspondant aux configurations de votre package.

Vous créez d'abord une chaîne, puis vous ajoutez vos sources VOD et vos sources en direct à la programmation de la chaîne en créant des programmes. Chaque programme est associé à une source VOD ou à une source en direct.

Rubriques

- [Création d'une chaîne à l'aide de la MediaTailor console](#)
- [Utilisation de groupes de sources avec les sorties de votre chaîne](#)
- [Supprimer une chaîne à l'aide de la MediaTailor console](#)

Création d'une chaîne à l'aide de la MediaTailor console

La procédure suivante décrit comment créer un canal à l'aide de la MediaTailor console.

Pour créer un canal

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Canaux.
3. Dans la barre de navigation, choisissez Create channel.
4. Sous Détails de la chaîne, entrez les informations relatives à votre chaîne :
 - Nom : Entrez le nom de votre chaîne.
 - Niveau : le niveau détermine les fonctionnalités prises en charge par la chaîne et le coût de son exploitation. Pour plus d'informations sur la tarification, consultez la [page de tarification de Channel Assembly](#). MediaTailor prend en charge les niveaux suivants :
 - Basique : le niveau Basic prend en charge les modes de lecture linéaire et en boucle, mais ne prend pas en charge les sources en direct.
 - Standard - Le niveau Standard prend en charge les sources en direct et nécessite le mode de lecture linéaire.

Lorsque vous sélectionnez Standard dans les détails de la chaîne, vous pouvez définir les audiences sous Détails des audiences. Ces audiences seront utilisées pour ProgramRules lorsque vous créerez AudienceMedia pour votre programme par défaut.

- Choisissez Ajouter.
- Entrez le nom du public dans la zone de texte. Il doit comporter entre 1 et 32 caractères alphanumériques.
- Choisissez Confirmer.

- Choisissez Suivant.

- Mode de lecture : le mode de lecture définit le comportement de lecture de la chaîne.

MediaTailor prend en charge les modes de lecture suivants :

- Boucle - Les programmes du programme se déroulent back-to-back en boucle sans fin. Une fois le dernier programme lancé dans un programme, la lecture revient en boucle au premier programme. La lecture continue en boucle jusqu'à ce que vous arrêtiez la chaîne.
- Linéaire - Chaque programme du calendrier est joué une fois, back-to-back.

5.

Pour Filler Slate, sélectionnez le nom de l'emplacement source faisant référence à l'emplacement de l'ardoise, et le nom de la source VOD à utiliser comme ardoise. MediaTailor utilise l'ardoise pour combler les lacunes entre les programmes du calendrier. Si la durée de l'ardoise est inférieure à la durée de l'écart entre les programmes, MediaTailor boucle l'ardoise. Vous devez configurer le champ Filler Slate si votre chaîne utilise le mode de lecture linéaire. MediaTailor ne prend pas en charge le mode de lecture en boucle.

6. Choisissez Suivant.

7. Spécifiez les détails de l'audience dans les règles du programme.

8. Lorsque vous sélectionnez Standard dans les détails de la chaîne, vous pouvez définir les audiences sous Détails des audiences. Ces audiences seront utilisées pour ProgramRules lorsque vous créerez AudienceMedia pour votre programme par défaut :

- Sélectionnez Ajouter, puis ajoutez une audience dans la zone de texte, puis sélectionnez Confirmer.

 Note

Entrez un nom ne dépassant pas_ 32 caractères alphanumériques.

- Type de sortie : sélectionnez le format de diffusion de la chaîne. DASH et HLS sont pris en charge.
- Groupe source : entrez le nom du groupe source que vous avez créé dans la configuration de votre package, comme décrit dans [Ajouter des sources VOD à votre emplacement source](#).

9. Sélectionnez Suivant.

10. Sous Paramètres du manifeste, entrez des informations supplémentaires sur les paramètres du manifeste :

- Fenêtre de manifeste (sec) : fenêtre temporelle, en secondes, contenue dans chaque manifeste. La valeur minimale est de 30 secondes et la valeur maximale de 3 600 secondes.
 - Type de balisage publicitaire (sorties HLS uniquement) : type de balises publicitaires qui apparaissent dans les pauses publicitaires des programmes de VOD. Sélectionnez cette option Daterange pour MediaTailor insérer des pauses publicitaires dans les programmes de VOD avec des EXT-X-DATERANGE tags. Sélectionnez cette option Scte35 Enhanced pour MediaTailor insérer des pauses publicitaires dans les programmes de VOD à l'aide de EXT-X-CUE-IN tags EXT-X-CUE-OUT et. Pour plus d'informations sur ces types de balises, consultez [SCTE-35 messages pour les pauses publicitaires](#). Pour les flux de travail en direct, passe MediaTailor toujours par les DATERANGE balises et ne passe par aucune balise Scte35 améliorée, quel que soit le type de balisage publicitaire sélectionné.
11. Si vous souhaitez configurer des sorties à canaux multiples, sous Sorties, choisissez Ajouter. Configurez ensuite les détails de votre sortie en effectuant les étapes 6 et 7 de cette procédure.
 12. Choisissez Suivant.
 13. Sous Politique de chaîne, choisissez les paramètres de politique IAM de votre chaîne :
 - Politique relative à l'interdiction de joindre une chaîne : limitez la diffusion aux seules personnes ayant accès aux informations d'identification de ce compte.
 - Joindre une politique personnalisée : définissez votre propre politique et limitez l'accès au plus petit nombre ou au plus grand nombre que vous le souhaitez.
 - Attachez une politique publique : acceptez toutes les demandes entrantes des clients à la sortie d'une chaîne. Vous devez utiliser cette option si vous souhaitez utiliser l'insertion d' MediaTailor annonces.
 14. Choisissez Suivant.
 15. Vérifiez vos paramètres dans le volet Révision et création.
 16. Choisissez Create channel (Créer un canal).

 Note

Les canaux sont créés dans un état arrêté. Votre chaîne ne sera active que lorsque vous l'aurez démarrée avec la MediaTailor console ou l' MediaTailor StartChannel API.

Utilisation de groupes de sources avec les sorties de votre chaîne

Un groupe de sources associe une configuration de package à une sortie sur un canal. Lorsque vous créez la configuration du package sur la source, vous identifiez le nom du groupe source. Ensuite, lorsque vous créez la sortie sur le canal, vous entrez le même nom pour associer la sortie à la configuration du package. Les sources VOD et les sources en direct ajoutées à un programme sur une chaîne doivent appartenir au groupe de sources identifié dans la sortie.

Par exemple :

- Les sources VOD 1 et 2 ont toutes deux trois configurations de package contenant les groupes de sources : HLS, DASH et HLS-4k.
- La source 3 de VOD possède deux configurations de paquets avec les groupes de sources HLS et DASH.

Si le canal A possède deux sorties avec les groupes de sources HLS et DASH, la sortie du canal peut utiliser les trois sources VOD. C'est parce que les sources VOD 1, 2 et 3 ont toutes des configurations de packages avec les labels de groupe de sources HLS et DASH.

Si le canal B possède deux sorties avec les groupes de sources HLS et HLS-4k, il peut utiliser les sources VOD 1 et 2, mais pas 3. Cela est dû au fait que les sources VOD 1 et 2 ont toutes deux des configurations de package avec les labels de groupe de sources HLS et HLS-4k.

Si le canal C possède une seule sortie avec le groupe de sources DASH, il peut utiliser les trois sources VOD. Les trois sources de VOD ont des configurations de package avec le groupe de sources DASH.

Supprimer une chaîne à l'aide de la MediaTailor console

Pour supprimer votre chaîne, procédez comme suit.

Pour supprimer votre chaîne

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Canaux.
3. Sélectionnez la chaîne que vous souhaitez supprimer.

4. Si votre chaîne est active, dans le menu déroulant Actions, sélectionnez Arrêter. Vous devez arrêter votre chaîne avant de pouvoir la supprimer.
5. Lorsque votre chaîne est arrêtée, dans le menu déroulant Actions, choisissez Supprimer.

Ajouter un programme à la programmation d'une chaîne

Chaque programme contient une source VOD ou une source en direct qui fait partie d'un emplacement source de votre compte. Vous ajoutez vos programmes à la programmation de votre chaîne pour contrôler l'ordre dans lequel ils sont diffusés dans le stream de votre chaîne.

Un programme contenant une source de VOD peut être configuré avec un ou plusieurs pauses publicitaires. Chaque pause publicitaire contient une ardoise, qui est une source de VOD provenant d'un emplacement source. Pour créer le break publicitaire, vous ajoutez l'ardoise à un décalage de quelques millisecondes au début du programme.

Rubriques

- [Création d'un programme dans le cadre d'un planning de chaînes à l'aide de la MediaTailor console](#)
- [Définition des cohortes d'audience et du contenu alternatif à l'aide des règles du programme](#)
- [Génération de manifestes spécifiques au public](#)

Création d'un programme dans le cadre d'un planning de chaînes à l'aide de la MediaTailor console

La procédure suivante explique comment créer un programme dans le planning de votre chaîne à l'aide de la MediaTailor console. Il décrit également comment configurer les pauses publicitaires, qui sont facultatives. Pour plus d'informations sur la création de programmes à l'aide de l' API MediaTailor, consultez [CreateProgram](#) le Guide de référence des AWS Elemental MediaTailor API.

Pour ajouter un programme

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Canaux.
3. Dans le volet Canaux, choisissez le canal que vous avez créé dans la [Pour créer un canal](#) procédure.
4. Dans la section Détails du programme, saisissez les détails de votre programme :

- Nom : il s'agit du nom du programme que vous ajoutez à votre chaîne.
- Type de source : Détermine le type de vidéo source diffusé par le programme. Cette option n'est disponible que pour les chaînes standard.
 - VOD - Le programme diffuse une video-on-demand source, telle qu'un épisode télévisé préenregistré.
 - En direct - Le programme diffuse une source en direct, telle qu'une émission d'actualités en direct.
- Nom de l'emplacement de la source : l'emplacement de la source MediaTailor associé au programme.
 - Si vous choisissez Sélectionner un emplacement source existant, choisissez un nom d'emplacement source dans le menu Sélectionnez un emplacement source. Vous pouvez également rechercher l'emplacement de votre source par son nom. Cela est utile si vous disposez d'un grand nombre d'emplacements sources.
 - Si vous choisissez Entrez le nom de l'emplacement source, recherchez votre emplacement source par son nom.
- Nom de la source VOD : nom de la source VOD associée MediaTailor au programme :
 - Si vous choisissez Sélectionner une source VOD existante, sélectionnez un nom de source VOD dans la liste des sources VOD associées à votre compte. Vous pouvez également rechercher votre source de VOD par son nom. C'est utile si vous disposez d'un grand nombre de sources de VOD.
 - Si vous choisissez Rechercher par nom, recherchez votre source en direct par son nom.
- Nom de la source en direct : nom de la source en direct à associer au programme. Cette option n'est disponible que si vous avez sélectionné Live comme type de source.
 - Si vous choisissez Sélectionner un emplacement source existant, choisissez un nom d'emplacement source dans le menu Sélectionnez un emplacement source. Vous pouvez également rechercher l'emplacement de votre source par son nom. Cela est utile si vous disposez d'un grand nombre d'emplacements sources.
 - Si vous choisissez Entrez le nom de l'emplacement source, recherchez votre emplacement source par son nom.
- Nom de la source VOD : nom de la source VOD associée MediaTailor au programme :
 - Si vous choisissez Sélectionner une source en direct existante, sélectionnez un nom de source en direct dans la liste des sources en direct associées à votre compte. Vous

pouvez également rechercher votre source en direct par son nom. Cela est utile si vous disposez d'un grand nombre de sources en direct.

- Si vous choisissez Rechercher par nom, recherchez votre source en direct par son nom.

5. Sélectionnez Suivant pour accéder à l'onglet Configuration du calendrier.

6. Sous Configuration de la lecture, définissez le moment où un programme est diffusé dans le calendrier de votre chaîne :

- **Durée en millisecondes** : définit la durée du programme en millisecondes. Cette option n'est disponible que pour les programmes utilisant des sources dynamiques.
- **Type de transition** : définit les transitions d'un programme à l'autre dans le planning :
 - **Relatif** : le programme est diffusé avant ou après un autre programme du programme. Cette option n'est disponible que pour les programmes utilisant des sources de VOD.
 - **Absolu** : le programme s'exécute à une heure précise de l'horloge murale. MediaTailor fait de son mieux pour exécuter le programme à l'heure que vous spécifiez. MediaTailor lance la lecture du programme sur une limite de segment commune entre le programme ou l'ardoise précédent. Cette option n'est disponible que pour les canaux configurés pour utiliser le linéaire. [Mode de lecture : le mode de lecture définit le comportement de lecture de la chaîne. MediaTailor prend en charge les modes de lecture suivants](#) :.
- **Heure de début du programme** : pour les types de transition absolus, heure de l'horloge murale à laquelle la lecture du programme est programmée. Si vous ajoutez ce programme à un canal linéaire en cours d'exécution, vous devez entrer une heure de début correspondant à 15 minutes ou plus tard par rapport à l'heure actuelle.
- **Position relative** : Choisissez où insérer le programme dans le planning, par rapport à un autre programme. Vous pouvez sélectionner Avant le programme ou Après le programme. Ce paramètre ne s'applique pas s'il s'agit de la première émission de la programmation de votre chaîne.
 - Si vous choisissez Sélectionner un programme existant, sélectionnez le nom du programme dans une liste prédéfinie des 100 prochains programmes diffusés par la chaîne dans le menu Utiliser le programme existant.
 - Si vous choisissez Rechercher un programme par son nom, entrez le nom d'un programme existant dans votre chaîne.

Si vous souhaitez ajouter des pauses publicitaires à votre programme, passez à l'étape suivante. Les pauses publicitaires ne sont configurables que pour les programmes utilisant des sources de VOD. Pour les sources en direct, les pauses publicitaires dans les manifestes DASH et les pauses publicitaires dans les manifestes HLS qui utilisent la EXT-X-DATERANGE balise sont transmises automatiquement.

7. Sélectionnez Suivant pour accéder à Ajouter des pauses publicitaires.
8. Sélectionnez Ajouter un break publicitaire. Sous Interruptions publicitaires, configurez les paramètres des pauses publicitaires :
 - Nom de l'emplacement source de Slate : Choisissez Sélectionnez un emplacement source existant et choisissez l'emplacement source où est stockée votre ardoise que vous avez créé plus tôt dans cette tâche.
 - Nom de la source VOD : choisissez Sélectionnez une source VOD existante et choisissez la source VOD que vous utilisez pour la liste que vous avez ajoutée plus tôt dans cette tâche. La durée de la liste détermine la durée de la pause publicitaire.
 - Décalage en millisecondes : cette valeur détermine l'heure de début de la pause publicitaire en millisecondes, sous forme de décalage par rapport au début du programme. Entrez une valeur inférieure à la durée de la source VOD et correspondant à une limite de segment sur toutes les pistes de la source VOD du programme (toutes les pistes audio, vidéo et sous-titrées), sinon le break publicitaire sera ignoré. Par exemple, si vous entrez 0, cela crée une pause publicitaire préalable à la diffusion qui est diffusée avant le début du programme.
 - Numéro de disponibilité : [MediaTailor écrit cette valeur est écritesplice_insert.avail_num, comme défini dans la section 9.7.3.1 de la spécification SCTE-35, Digital Program Insertion Cueing Message](#). La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256 inclus.
 - Disponibilité attendue : MediaTailor écrit cette valeur dans `splice_insert.avails_expected`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256 inclus.
 - ID d'événement Splice : MediaTailor écrit cette valeur dans `splice_insert.splice_event_id`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 1.
 - ID de programme unique : MediaTailor écrit cette valeur dans `splice_insert.unique_program_id`, comme défini dans la section 9.7.3.1 de la

spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256 inclus.

9. Pour un canal linéaire standard, sélectionnez Suivant pour passer à Définir un support alternatif.

Pour plus d'informations sur MediaTailor l'utilisation de supports alternatifs, consultez [Création de médias alternatifs](#).

Pour obtenir des informations plus avancées sur l'utilisation MediaTailor de la personnalisation de vos pauses publicitaires, consultez [Insérez des publicités et des pauses publicitaires personnalisées dans le flux d'une chaîne](#).

10. Sélectionnez Suivant pour accéder à Réviser et créer.

11. Sélectionnez Ajouter un programme.

Pour obtenir des informations plus avancées sur l'utilisation MediaTailor de la personnalisation de vos pauses publicitaires, consultez [Insérez des publicités et des pauses publicitaires personnalisées dans le flux d'une chaîne](#).

- 12.

 Important

Pour les chaînes en boucle, si vous modifiez la liste des programmes d'un programme programmé dans les 10 prochaines minutes, la modification ne sera visible qu'à la prochaine boucle.

Sous Détails du programme, entrez les détails de votre programme :

- Nom : il s'agit du nom du programme que vous ajoutez à votre chaîne.
- Type de source : détermine le type de source diffusé par le programme. Cette option n'est disponible que pour les chaînes standard.
 - VOD - Le programme diffuse une source VOD, telle qu'un épisode télévisé préenregistré.
 - En direct - Le programme diffuse une source en direct, telle qu'une émission d'actualités en direct.
- Nom de l'emplacement de la source : l'emplacement de la source à associer au programme.

Si vous choisissez Sélectionner un emplacement source existant, sélectionnez un nom d'emplacement source dans le menu déroulant Sélectionnez un emplacement source. Vous pouvez également rechercher l'emplacement de votre source par son nom. Cela est utile si vous disposez d'un grand nombre d'emplacements sources.

Si vous choisissez Entrez le nom de l'emplacement source, recherchez votre emplacement source par son nom.

- Nom de la source VOD : nom de la source VOD à associer au programme.

Si vous choisissez Sélectionner une source VOD existante, sélectionnez un nom de source VOD dans la liste des sources VOD associées à votre compte. Vous pouvez également rechercher votre source de VOD par son nom. C'est utile si vous disposez d'un grand nombre de sources de VOD.

Si vous choisissez Rechercher par nom, recherchez votre source de VOD par son nom.

- Nom de la source en direct : nom de la source en direct à associer au programme. Cette option n'est disponible que si vous avez sélectionné Live comme type de source.

Si vous choisissez Sélectionner une source en direct existante, sélectionnez un nom de source en direct dans la liste des sources en direct associées à votre compte. Vous pouvez également rechercher votre source en direct par son nom. Cela est utile si vous disposez d'un grand nombre de sources en direct.

Si vous choisissez Rechercher par nom, recherchez votre source en direct par son nom.

13. Sous Configuration de la lecture, définissez le moment où un programme est diffusé dans le calendrier de votre chaîne :

- Durée en millisecondes : définit la durée du programme en millisecondes. Cette option n'est disponible que pour les programmes utilisant des sources dynamiques.
- Type de transition : définit les transitions d'un programme à l'autre dans le planning.
 - Relatif - Le programme est diffusé avant ou après un autre programme du programme. Cette option n'est disponible que pour les programmes utilisant des sources de VOD.
 - Absolu - Le programme est diffusé à une heure précise de l'horloge murale. MediaTailor fait de son mieux pour exécuter le programme à l'heure que vous spécifiez. Nous commençons la lecture du programme sur une limite de segment commune entre le programme ou la liste précédente. Cette option n'est disponible que pour les canaux configurés pour utiliser le [linear playback mode](#).

 Note

Tenez compte du comportement suivant pour les types de transition absolus :

- Si le programme précédent du programme a une durée qui s'étend au-delà de l'heure de l'horloge murale, MediaTailor tronque le programme précédent sur la limite du segment commun la plus proche de l'heure de l'horloge murale.
- S'il y a des écarts entre les programmes dans le calendrier, MediaTailor joue [filler slate](#). Si la durée de l'ardoise est inférieure à la durée de l'écart, MediaTailor enrroulez l'ardoise.

- Heure de début du programme - Pour les types de transition absolus, heure murale à laquelle la lecture du programme est programmée. Si vous ajoutez ce programme à un canal linéaire en cours d'exécution, vous devez entrer une heure de début correspondant à 15 minutes ou plus tard par rapport à l'heure actuelle.
- Position relative : Choisissez où insérer le programme dans le planning par rapport à un autre programme. Vous pouvez sélectionner Avant le programme ou Après le programme. Ce paramètre ne s'applique pas s'il s'agit de la première émission de la programmation de votre chaîne.
- Programme relatif : nom du programme à utiliser pour insérer le nouveau programme avant ou après. Ce paramètre ne s'applique pas s'il s'agit de la première émission de la programmation de votre chaîne.

Si vous choisissez Sélectionner un programme existant, sélectionnez le nom du programme dans une liste prédéfinie des 100 prochains programmes diffusés par la chaîne dans le menu déroulant Utiliser le programme existant.

Si vous choisissez Rechercher un programme par son nom, entrez le nom d'un programme existant dans votre chaîne.

Si vous souhaitez ajouter des pauses publicitaires à votre programme, passez à l'étape suivante. Les pauses publicitaires ne sont configurables que pour les programmes utilisant des sources de VOD. Pour les sources en direct, les pauses publicitaires dans les manifestes DASH et les pauses publicitaires dans les manifestes HLS qui utilisent la EXT-X-DATERANGE balise sont transmises automatiquement.

14. Sélectionnez Ajouter un break publicitaire. Sous Interruptions publicitaires, configurez les paramètres des pauses publicitaires :

- Nom de l'emplacement source de Slate : Choisissez Sélectionnez un emplacement source existant et choisissez l'emplacement source où est stockée votre ardoise que vous avez créé plus tôt dans ce didacticiel.
- Nom de la source VOD : Choisissez Sélectionnez une source VOD existante et choisissez la source VOD que vous utilisez pour Slate que vous avez ajoutée plus tôt dans ce didacticiel. La durée de la liste détermine la durée de la pause publicitaire.
- Pour le décalage en millisecondes : cette valeur détermine l'heure de début de la pause publicitaire en millisecondes, sous forme de décalage par rapport au début du programme. Entrez une valeur inférieure à la durée de la source VOD et correspondant à une limite de segment sur toutes les pistes de la source VOD du programme (toutes les pistes audio, vidéo et sous-titrées), sinon le break publicitaire sera ignoré. Par exemple, si vous entrez 0, cela crée une pause publicitaire préalable à la diffusion qui est diffusée avant le début du programme.

 Note

S'il MediaTailor détecte des marqueurs publicitaires, tels que DATERANGE ou EXT-X-CUE-OUT pour HLS et EventStream pour DASH, avec une durée nulle dans votre source de VOD, vous pouvez sélectionner le décalage de ces marqueurs publicitaires dans le menu déroulant à utiliser comme décalage du break publicitaire. Pour qu'une opportunité publicitaire soit détectée, elle doit être présente au même décalage dans toutes les configurations de package d'une source VOD, et sa durée doit être nulle.

- Type de message : type d'insertion publicitaire SCTE-35. Choisissez SPLICE_INSERT ou TIME_SIGNAL :
 - SPLICE_INSERT : fournit des métadonnées de base sur le break publicitaire à l'aide des paramètres Splice Insert.
 - TIME_SIGNAL : fournit des métadonnées plus avancées à l'aide de descripteurs de segmentation. Pour plus d'informations sur les différences entre les types de messages, consultez [the section called “SCTE-35 messages pour les pauses publicitaires”](#).

Pour le type de message SPLICE_INSERT :

- Pour le numéro Avail, il est écrit `splice_insert.avail_num`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256, inclusives.

- Pour l'Avail attendu, ceci est écrit `splice_insert.avails_expected`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256, inclusives.
- Pour l'identifiant d'événement Splice, celui-ci est écrit `splice_insert.splice_event_id`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 1.
- Pour l'ID de programme unique, celui-ci est écrit `splice_insert.unique_program_id`, comme défini dans la section 9.7.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256, inclusives.

Pour le type de message TIME_SIGNAL :

- Pour l'ID d'événement de segmentation, celui-ci est écrit `segmentation_descriptor.segmentation_event_id`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur par défaut est 1.
- Pour l'ID de type de segmentation, celui-ci est écrit `segmentation_descriptor.segmentation_type_id`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur par défaut est 48 (0x30). Les valeurs doivent être comprises entre 0 et 256, inclusives.
- Pour l'UPID de segmentation, celui-ci est écrit `segmentation_descriptor.segmentation_upid`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur doit être une chaîne hexadécimale contenant les caractères 0-9 et A-F. La valeur par défaut est une chaîne vide.
- Pour le type UPID de segmentation, celui-ci est écrit `segmentation_descriptor.segmentation_upid_type`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur par défaut est 14 (0x0E). Les valeurs doivent être comprises entre 0 et 256, inclusives.
- Pour le numéro de segment, il est écrit `segmentation_descriptor.segment_num`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256, inclusives.
- Pour les segments attendus, ceci est écrit `segmentation_descriptor.segments_expected`, comme défini dans la section 10.3.3.1 de la spécification SCTE-35. La valeur par défaut est 0. Les valeurs doivent être comprises entre 0 et 256, inclusives.

15. Choisissez Ajouter un programme.

Pour obtenir des informations plus avancées sur MediaTailor la personnalisation de vos pauses publicitaires, consultez [Insérez des publicités et des pauses publicitaires personnalisées dans le flux d'une chaîne](#).

 Note

Si votre chaîne comporte au moins une sortie avec un type de balisage Enhanced SCTE35 publicitaire, vous pouvez envoyer des métadonnées d'annonce publicitaire. MediaTailor écrit les paires clé-valeur soumises dans le EXT-X-ASSET tag pour votre pause publicitaire.

Définition des cohortes d'audience et du contenu alternatif à l'aide des règles du programme

Avec les règles du programme, vous pouvez définir des groupes d'audience pour une chaîne et spécifier des médias alternatifs à diffuser pour ces publics. Vous pouvez associer une ou plusieurs sources de contenu alternatives à une audience pour un programme. Une fois le programme terminé, le contenu public par défaut sera diffusé, sauf si vous spécifiez d'autres médias alternatifs.

Les règles du programme sont disponibles sur les canaux de niveau STANDARD avec le mode de lecture LINEAR. MediaTailor les chaînes prennent en charge les médias alternatifs pour toutes les sources de VOD et les sources en direct.

Pour un exemple d'utilisation, consultez la section [Utilisation des règles du programme avec AWS MediaTailor](#).

Définition des publics

Définissez les audiences d'une chaîne en saisissant une audience une par une lors de la configuration d'une MediaTailor chaîne. Vous pouvez le faire par le biais de la MediaTailor console ou de l' `MediaTailorCreateChannelAPI`. Chaque audience doit comporter entre 1 et 32 caractères alphanumériques. Si les valeurs fournies pour les audiences ne sont pas valides, la demande échoue.

Vous ne pouvez définir des audiences que sur les chaînes de niveau STANDARD avec le mode de lecture LINEAR.

Lorsque vous devez mettre à jour les audiences, vous pouvez le faire à l'aide de la MediaTailor console ou de l' `MediaTailor UpdateChannelAPI`.

Si vous utilisez `ProgramRules` cette fonctionnalité, assurez-vous que l'audience `AudienceMedia` définie dans `CreateProgram` ou dans la `UpdateProgram` demande contient l'audience existante définie dans la chaîne.

Création de médias alternatifs

La tâche suivante explique comment définir un autre média à l'aide de la MediaTailor console. Pour plus d'informations sur la façon de définir des médias alternatifs à l'aide de l' `MediaTailor API`, consultez [CreateProgram](#) la référence de l'AWS Elemental MediaTailor API.

Pour définir un autre média dans un nouveau programme :

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, sélectionnez Assemblage de canaux > Canaux.
3. Sélectionnez le nom de la chaîne à laquelle vous souhaitez ajouter un autre média.
4. Créez un programme. Pour de plus amples informations, veuillez consulter [Création d'un programme dans le cadre d'un planning de chaînes à l'aide de la MediaTailor console](#).
5. Configurez les médias alternatifs :
 - Sélectionnez Ajouter dans la zone Audiences pour sélectionner le public pour lequel vous définissez des médias alternatifs.
 - Sélectionnez une audience définie sur la chaîne dans le menu Audience.
 - Sélectionnez Ajouter un média de remplacement pour commencer à définir un autre média pour le programme.
 - MediaTailor crée une boîte Alternate media 1. Il s'agit du premier contenu diffusé MediaTailor en tant que média alternatif dans le programme.
 - Dans le champ Autre média 1 :
 - Sélectionnez un emplacement source.
 - Sélectionnez un type de VOD ou de source en direct :

Pour VOD

- Sélectionnez VOD comme type de source.
- (Facultatif) spécifiez une plage de clips. Avec les sources VOD, y compris les sources VOD de médias alternatifs, vous pouvez spécifier une partie d'une source VOD à lire, en

découpant du début and/or à la fin de la source. Spécifiez Les décalages de début et de fin sont exprimés en millisecondes.

- (Facultatif) a \ Ajouter des pauses publicitaires. Cela se fait de la même manière que lors de la création de programmes. Pour de plus amples informations, veuillez consulter [Création d'un programme dans le cadre d'un planning de chaînes à l'aide de la MediaTailor console](#).

Pour Live

- Sélectionnez Live comme type de source.
- Sélectionnez une source en direct.
- Entrez une heure de début en millisecondes par rapport à l'heure à laquelle cette source en direct doit démarrer. La source en direct ne sera jouée que pendant la période du programme par défaut sur lequel elle est définie. Si l'heure de début est antérieure au démarrage du programme par défaut, celui-ci ne commencera pas tant que le programme par défaut ne le fera pas. Si l'heure de début est postérieure à la fin du programme par défaut, la source en direct ne MediaTailor sera pas diffusée.
- Entrez une durée en millisecondes. La durée doit être d'au moins 10 minutes.
- Des médias alternatifs supplémentaires peuvent être ajoutés à ce programme pour le public en sélectionnant à nouveau Ajouter un autre média. Cela créera une autre boîte intitulée Autre média 2. Vous pouvez spécifier jusqu'à 5 sources de médias alternatifs par programme et par public.
- Une fois que vous avez fini de définir les médias alternatifs pour tous les publics souhaités, sélectionnez Suivant et poursuivez la création du programme.

Pour de plus amples informations, veuillez consulter [Création d'un programme dans le cadre d'un planning de chaînes à l'aide de la MediaTailor console](#).

Note

Les médias alternatifs ne sont diffusés que pendant la période du programme sur lequel ils sont définis. Si tout le contenu alternatif dépasse le contenu par défaut, il MediaTailor sera tronqué. MediaTailor lit un autre média dans l'ordre dans lequel il est défini. Les heures de démarrage des médias alternatifs en direct seront toujours prioritaires et tronqueront les sources VOD ou les sources en direct précédemment

programmées. Tout moment qui n'est pas rempli de médias alternatifs pour un public sera rempli avec la liste de diffusion définie par la chaîne

- Pour définir les médias d'audience destinés à d'autres publics, sélectionnez à nouveau Ajouter à côté de Audiences. Sélectionnez l'audience nouvellement créée, définissez l'identifiant de l'audience et ajoutez un autre média comme décrit ci-dessus. Jusqu'à 5 spectateurs peuvent avoir accès à des médias alternatifs pour chaque émission.

Génération de manifestes spécifiques au public

Pour récupérer un manifeste destiné à un public particulier, utilisez le paramètre de `aws.mediatailor.channel.audienceId` requête. Ce paramètre de requête peut être ajouté dynamiquement par votre CDN ou ajouté via un appel à votre contenu ou à votre système de gestion des clients. Vous devez conserver l'association d'une session de lecture donnée à un utilisateur `audienceId` externe à MediaTailor. Cela permettra de récupérer un manifeste spécifique au public avec tout média alternatif défini pour ce public à la place du contenu par défaut. Il est important qu'une fois qu'un manifeste est demandé pour un public en particulier, le joueur demande toujours le manifeste avec le même identifiant d'audience, sinon des erreurs de lecture pourraient se produire.

Si une demande est faite pour une audience qui n'existe pas sur la chaîne, MediaTailor renvoie une erreur 404.

Exemple Obtenir un manifeste pour un public

```
https:// prefix>.channel-assembly.mediatailor.us-  
west-2.amazonaws.com/v1/channel/ExampleChannel/index_dash.mpd?  
aws.mediatailor.channel.audienceId=Seattle
```

Insérez des publicités et des pauses publicitaires personnalisées dans le flux d'une chaîne

Vous pouvez ainsi monétiser les flux linéaires d'assemblage de chaînes en insérant des pauses publicitaires dans vos programmes sans conditionner le contenu à l'aide MediaTailor de marqueurs. SCTE-35 Vous pouvez utiliser l'assemblage de chaînes avec le service d'insertion de MediaTailor publicités ou avec n'importe quelle insertion d'annonces côté serveur (SSAI).

Les rubriques suivantes expliquent comment insérer des publicités et des breaks publicitaires personnalisés dans le flux linéaire de votre chaîne.

Rubriques

- [Configuration de l'insertion d'annonces avec MediaTailor](#)
- [SCTE-35 messages pour les pauses publicitaires](#)

Configuration de l'insertion d'annonces avec MediaTailor

Pour insérer des publicités personnalisées dans le flux de votre chaîne, l'URL du point de terminaison de votre chaîne est la source de contenu pour AWS Elemental MediaTailor. Ce guide explique comment configurer l'insertion MediaTailor d'annonces.

Conditions préalables

Avant de commencer, vérifiez que vous respectez les conditions requises suivantes :

- Préparez vos flux HLS et DASH pour l'insertion de MediaTailor publicités.
 - Si vous n'avez pas encore préparé de flux de contenu, consultez [Étape 2 : Préparer un stream](#) la rubrique Commencer à insérer des MediaTailor annonces.
- Disposez d'un serveur de décision publicitaire (ADS).
- Configurez les paramètres Ad Break dans le programme. Pour plus d'informations, consultez la [Configuring ad breaks for your program](#) procédure.

Il est recommandé d'utiliser un réseau de diffusion de contenu (CDN) entre l'assemblage des chaînes et l'insertion de MediaTailor publicités. Le service d'insertion d' MediaTailor annonces peut générer des demandes d'origine supplémentaires. Il est donc recommandé de configurer votre CDN pour qu'il utilise par proxy les manifestes issus de l'assemblage des canaux, puis d'utiliser les URL préfixées du CDN sur l'URL de la source de contenu.

Configuration MediaTailor pour l'insertion d'annonces

Vous trouverez ci-dessous comment configurer les paramètres de MediaTailor la console afin de pouvoir insérer des publicités personnalisées dans le stream de votre chaîne.

Pour configurer MediaTailor l'insertion d'annonces

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, sélectionnez Configurations.

3. Sous Paramètres obligatoires, entrez les informations de base requises concernant votre configuration :
 - Nom : nom de votre configuration.
 - Source du contenu : entrez l'URL de diffusion depuis la sortie de votre chaîne, sans le nom et l'extension du fichier. Pour des informations avancées sur MediaTailor la configuration, consultez [Réglages requis](#).
 - Serveur de décision publicitaire : entrez l'URL de votre ADS.
4. Vous pouvez éventuellement configurer les alias de configuration, les détails de personnalisation et les paramètres avancés. Pour plus d'informations sur ces paramètres, consultez [Paramètres de configuration facultatifs](#).
5. Dans la barre de navigation, choisissez Créer une configuration.

Maintenant que vous avez configuré l'insertion MediaTailor d'annonces, vous pouvez également configurer des pauses publicitaires. Pour obtenir des instructions complètes, consultez [Commencer à insérer MediaTailor des publicités](#).

SCTE-35 messages pour les pauses publicitaires

Avec MediaTailor, vous pouvez créer une chaîne de contenu en fonction de l'emplacement de la source et des ressources sources de VOD. Vous pouvez ensuite configurer un ou plusieurs pauses publicitaires pour chacun des programmes du calendrier d'une chaîne. Vous utilisez des messages basés sur les SCTE-35 spécifications pour conditionner le contenu pour les pauses publicitaires. Par exemple, vous pouvez utiliser SCTE-35 des messages pour fournir des métadonnées relatives aux pauses publicitaires. Pour plus d'informations sur la SCTE-35 spécification, voir [Message de repérage d'insertion de programme numérique](#).

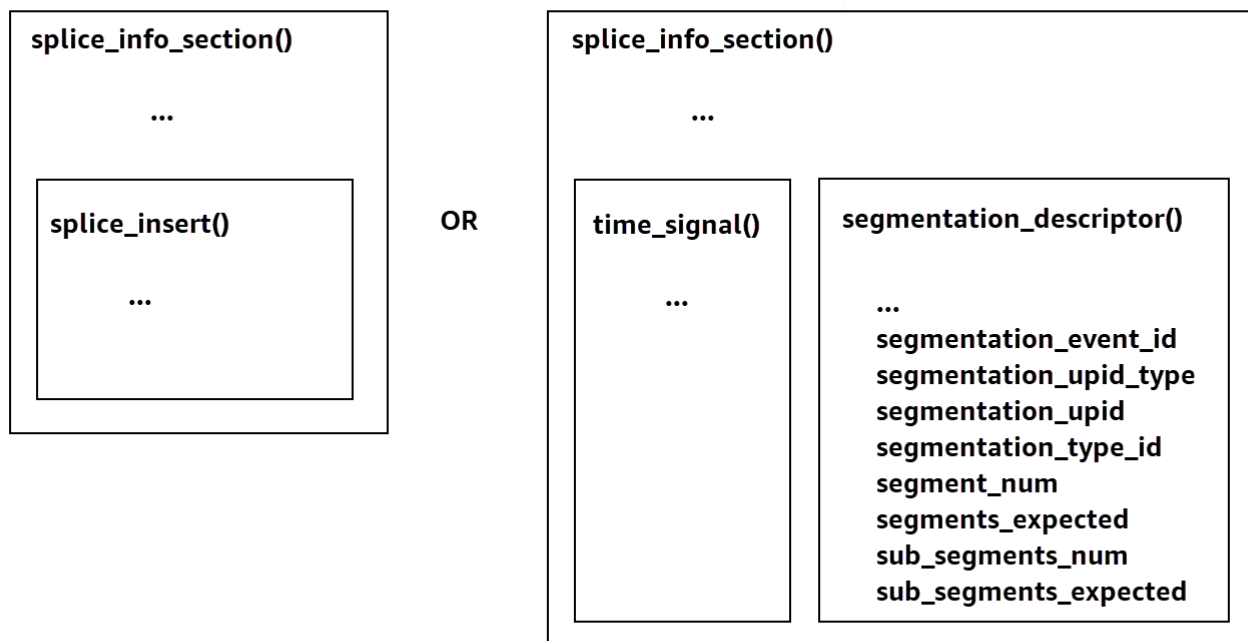
Vous pouvez configurer les pauses publicitaires de l'une des deux manières suivantes :

- Joindre un `time_signal` SCTE-35 message à un `segmentation_descriptor` message. Ce `segmentation_descriptor` message contient des champs de métadonnées plus avancés, tels que des identifiants de contenu, qui fournissent plus d'informations sur le break publicitaire. MediaTailor écrit les métadonnées de l'annonce dans le manifeste de sortie dans le cadre des SCTE-35 données du marqueur publicitaire EXT-X-DATERANGE EventStream (HLS) ou (DASH).
- Joindre un `splice_insert` SCTE-35 message fournissant des métadonnées de base sur le break publicitaire.
- HLS :

- Lorsque le type de balisage publicitaire est `Daterange`, MediaTailor spécifie les pauses publicitaires sous forme de `EXT-X-DATERANGE` balises dans le manifeste.
- Lorsque le type de balisage publicitaire est défini `Scte35 Enhanced`, MediaTailor spécifie les pauses publicitaires à l'aide des balises suivantes :
 - MediaTailor place un `EXT-X-CUE-OUT` sur le premier segment de la liste des annonces, indiquant une coupure entre le contenu et le break publicitaire. Il contient la durée prévue de la pause publicitaire, telle que `EXT-X-CUE-OUT:Duration=30`.
 - `>EXT-X-ASSET`: Cette balise apparaît sur le même segment que `EXT-X-CUE-OUT` les métadonnées publicitaires fournies `AdBreak` lors de la création ou de la mise à jour du programme et contient ces métadonnées. Il contient toujours `CAID`.
 - `EXT-0ATCLS-SCTE35`: cette balise apparaît sur le même segment que le message `EXT-X-CUE-OUT` et contient des octets codés en base64. `SCTE-35`
 - `EXT-X-CUE-OUT-CONT`: cette balise apparaît sur chaque segment suivant de la liste d'annonces et contient des informations sur la durée et le temps écoulé. Il contient également le `SCTE-35` message codé en base64 et le `CAID`
 - `EXT-X-CUE-IN`: cette balise apparaît sur le premier segment de contenu une fois la pause publicitaire terminée et indique une interruption entre une pause publicitaire et le retour au contenu.

L'illustration suivante montre les deux manières de configurer des pauses publicitaires dans une chaîne à l'aide de `SCTE-35` messages :

- Utilisez un `splice_insert()` message pour configurer des pauses publicitaires avec des métadonnées de base.
- Associez un `time_signal()` message à un `segmentation_descriptor()` message pour configurer des pauses publicitaires avec des métadonnées plus détaillées.



Pour plus d'informations sur l'utilisation `time_signal`, reportez-vous à la section 9.7.4 de la SCTE-35 spécification 2022, [Digital Program Insertion Gueing Message](#).

Les informations relatives aux interruptions publicitaires apparaissent dans les `splice_info_section` SCTE-35 données de sortie. Avec MediaTailor, vous pouvez associer un seul `segmentation_descriptor` message à un seul `time_signal` message.

Note

Si vous envoyez un `segmentation_descriptor` message, vous devez l'envoyer dans le cadre du type de `time_signal` message. Le `time_signal` message contient uniquement le `splice_time` champ MediaTailor construit à l'aide d'un horodatage donné.

Le tableau suivant décrit les champs MediaTailor obligatoires pour chaque `segmentation_descriptor` message. Pour plus d'informations, consultez la section 10.3.3.1 de la SCTE-35 spécification 2022, que vous pouvez acheter sur le site Web de l'[ANSI Webstore](#).

Champs obligatoires pour un message **segmentation_descriptor**

Champ	Type	Valeur par défaut	Description
segmentation_event_id	integer	1	Ceci est écrit poursegmentation_descriptor.segmentation_event_id .
segmentation_upid_type	entier	14 (0 x 0E)	Ceci est écrit poursegmentation_descriptor.segmentation_upid_type . La valeur doit être comprise entre 0 et 256 inclus.
segmentation_upid	chaîne	""(chaîne vide)	Ceci est écrit poursegmentation_descriptor.segmentation_upid . La valeur doit être une chaîne hexadécimale contenant les caractères 0-9 etA-F.
segmentation_type_id	entier	48 (0 x 30)	Ceci est écrit poursegmentation_descriptor.segmentation_type_id . La valeur doit être

Champ	Type	Valeur par défaut	Description
			comprise entre 0 et 256 inclus.
segment_num	entier	0	Ceci est écrit pour segmentat ion_descr iptor.segment_num . La valeur doit être comprise entre 0 et 256 inclus.
segments_ expected	entier	0	Ceci est écrit pour segmentat ion_descr iptor.segments_ expected . La valeur doit être comprise entre 0 et 256 inclus.
sub_segment_num	entier	null	Ceci est écrit pour segmentat ion_descr iptor.sub_ _segment_num . La valeur doit être comprise entre 0 et 256 inclus.

Champ	Type	Valeur par défaut	Description
sub_segments_expected	entier	null	Ceci est écrit pour segmentation_descriptor.sub_segments_expected. La valeur doit être comprise entre 0 et 256 inclus.

Le tableau suivant indique les valeurs MediaTailor définies automatiquement pour certains champs du segmentation_descriptor message.

Valeurs définies par MediaTailor pour les champs d'un message **segmentation_descriptor**

Champ	Type	Value
segmentation_event_cancel_indicator	Booléen	True
program_segmentation_flag	Booléen	True
delivery_not_restricted_flag	Booléen	True

MediaTailor définit toujours la valeur segmentation_duration_flag à True. MediaTailor remplit le segmentation_duration champ avec la durée, en ticks, du contenu de l'état.

Note

Lors de l'envoi time_signal des messages, le splice_command_type champ du splice_info_section message est défini sur 6 (0x06).

Dans la sortie HLS, pour un message AdBreak contenant un `time_signal` message, la EXT-X-DATERANGE balise de sortie inclut un SCTE-35 champ défini sur la version sérialisée du `splice_info_section` message. Par exemple, la EXT-X-DATERANGE balise suivante indique la version sérialisée du `splice_info_section` message :

```
#EXT-X-DATERANGE:ID="1",START-DATE="2020-09-25T02:13:20Z",DURATION=300.0,SCTE35-  
OUT=0xFC002C0000000000000000FF05068000000000001602144355454900000000100E000019BFCC00E00300000000
```

Dans la sortie DASH, pour un `time_signal` message AdBreak contenant un message, l'`EventStream`élément de sortie inclut un `scte35:SpliceInfoSection` élément `scte35:TimeSignal` et `scte35:SegmentationDescriptor` des éléments comme enfants. L'`scte35:TimeSignal`élément possède un `scte35:SpliceTime` élément enfant et l'`scte35:SegmentationDescriptor`élément possède un `scte35:SegmentationUpid` élément enfant. Par exemple, la sortie DASH suivante montre la structure de l'`EventStream`élément enfant de l'élément :

```
<EventStream schemaIdUri="urn:scte:sc35:2013:xml" timescale="90000">
  <Event duration="27000000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
      <scte35:TimeSignal>
        <scte35:SpliceTime ptsTime="0" />
      </scte35:TimeSignal>
      <scte35:SegmentationDescriptor segmentNum="0"
segmentationDuration="27000000" segmentationEventCancelIndicator="false"
segmentationEventId="1" segmentationTypeId="48" segmentsExpected="0">
        <scte35:SegmentationUpid segmentationUpidFormat="hexBinary"
segmentationUpidType="14">012345</scte35:SegmentationUpid>
      </scte35:SegmentationDescriptor>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
```

Vous avez appris à utiliser SCTE-35 les messages pour configurer des pauses publicitaires dans l'assemblage des chaînes, à définir la structure et les champs obligatoires de ces messages, ainsi qu'à obtenir des exemples de sorties HLS et DASH qui incluent les SCTE-35 messages.

Décalage temporel de lecture d'une chaîne

Grâce à l'assemblage des MediaTailor canaux, vous pouvez décaler la durée de lecture d'un canal. Le décalage horaire permet aux spectateurs de regarder le contenu d'une chaîne à un

moment différent de l'heure de diffusion initiale. Par exemple, un téléspectateur peut commencer à regarder une émission depuis le début, même s'il rejoint la diffusion après le début de celle-ci. Un téléspectateur peut également suspendre une émission et recommencer à la regarder plus tard.

Pour décaler la durée de lecture d'une chaîne, vous devez ajouter un `start` paramètre à l'URL de lecture de la chaîne. Le `start` paramètre indique l'heure à laquelle la lecture doit commencer, par rapport à l'heure actuelle. Par exemple, si l'heure actuelle est 14 h 00 et que vous souhaitez démarrer la lecture à 13 h 00, vous devez définir le `start` paramètre sur -1 heure en secondes. `-3600`

Le `start` paramètre accepte à la fois des valeurs positives et négatives, en secondes :

- Les valeurs négatives indiquent une époque passée par rapport à l'heure actuelle. Par exemple, `start=-3600` signifie « démarrer la lecture il y a 1 heure ».
- Les valeurs positives indiquent une date dans le futur, par rapport à l'heure actuelle. Par exemple, `start=3600` cela signifie « démarrer la lecture à partir d'une heure dans le futur ».

L'exemple suivant montre comment ajouter le `start` paramètre à l'URL de lecture d'une chaîne :

```
https://a1b2c3d4e5f6.mediapackage.us-west-2.amazonaws.com/out/v1/examplechannel/  
index.m3u8?start=-3600
```

Dans cet exemple, la lecture commence il y a 1 heure.

Le `start` paramètre est soumis aux limites suivantes :

- La fenêtre de décalage horaire maximale est de 24 heures dans le passé ou dans le futur.
- Le décalage temporel n'est disponible que pour les chaînes utilisant le mode de lecture linéaire.
- Le décalage temporel n'est pas disponible pour les chaînes qui utilisent le mode de lecture en boucle.

Si vous spécifiez un `start` paramètre situé en dehors de la fenêtre de décalage horaire disponible, MediaTailor renvoie une erreur.

Utilisation d'un CDN pour optimiser la personnalisation des MediaTailor publicités et la diffusion de contenu

AWS Elemental MediaTailor fonctionne efficacement en tant que service autonome, mais son intégration à un réseau de diffusion de contenu (CDN), tel qu'Amazon CloudFront ou un autre tiers CDNs, peut considérablement améliorer vos flux de travail de streaming. L'intégration d'un CDN est particulièrement utile lorsque vous devez diffuser du contenu à un large public géographiquement réparti ou lorsque vous souhaitez garantir une diffusion publicitaire cohérente dans différentes AWS régions.

Sans CDN, vos spectateurs se connectent directement à MediaTailor des manifestes et à des segments publicitaires personnalisés, ce qui peut entraîner une latence accrue, en particulier pour les spectateurs situés loin de la AWS région où votre MediaTailor configuration est déployée. En outre, lors d'événements à fort trafic, les connexions directes MediaTailor peuvent être soumises à une charge accrue, ce qui peut affecter les performances.

Pour plus d'informations sur les MediaTailor concepts et les flux de travail, consultez [Qu'est-ce que AWS Elemental MediaTailor?](#)

Lors de l'intégration d'un CDN MediaTailor, il est important de configurer une gestion CORS (Cross-Origin Resource Sharing) appropriée afin d'éviter les problèmes susceptibles de provoquer des échecs de lecture sur les lecteurs Web. Une configuration CORS appropriée est essentielle pour les segments publicitaires et les segments de contenu. Bien que les segments publicitaires soient plus sensibles aux problèmes de CORS, l'application d'une gestion cohérente du CORS à tous les types de segments garantit une expérience de lecture des plus fiables. Pour obtenir des instructions détaillées sur la configuration des comportements de routage des CDN avec une gestion CORS appropriée, consultez [Production-ready CloudFront configuration pour MediaTailor](#)

L'intégration du CDN permet également des fonctionnalités avancées de transmission de paramètres et de routage dynamique. Pour plus d'informations sur la transmission de paramètres de requête à CDNs des fins d'autorisation et de routage, consultez [MediaTailor paramètres de requête manifeste](#). Pour un serveur publicitaire dynamique et un routage d'origine à l'aide d'alias de configuration, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

Le fait de placer un CDN entre vos spectateurs MediaTailor offre les avantages suivants :

- Réduisez la latence en diffusant du contenu depuis des emplacements périphériques plus proches des spectateurs

- Améliorez l'évolutivité en répartissant la charge sur l'infrastructure mondiale du CDN
- Améliorez la fiabilité grâce à des voies de livraison redondantes
- Optimisez les coûts en réduisant le trafic d'origine
- Implémentez des fonctionnalités avancées telles que le routage adapté à la qualité des médias (MQAR) pour améliorer la qualité du streaming

Rubriques

- [Sélectionnez le CDN adapté à vos besoins](#)
- [Planifiez votre intégration au CDN pour AWS Elemental MediaTailor](#)
- [Configurer l'intégration du CDN avec MediaTailor](#)
- [Configurez SSAI avec un CDN pour une publicité vidéo personnalisée](#)
- [Créez des canaux MediaTailor linéaires avec assemblage de canaux et CDN](#)
- [Intégration MediaTailor avec un MediaPackage CDN](#)
- [Intégration AWS Elemental MediaTailor à Amazon CloudFront](#)
- [Configurer un tiers CDN pour la diffusion MediaTailor des publicités](#)
- [Guide d'optimisation des performances pour le CDN et MediaTailor les intégrations](#)
- [Surveillez les opérations et les performances du MediaTailor CDN](#)
- [Tests et validation pour le CDN et MediaTailor les intégrations](#)
- [Résoudre les problèmes liés à l'intégration MediaTailor du CDN](#)
- [Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor](#)
- [Automatisez MediaTailor et créez un CDN avec CloudFormation](#)
- [Production-ready CloudFront configuration pour MediaTailor](#)
- [Bénéficiez d'une assistance et d'une aide au dépannage pour le CDN et les intégrations MediaTailor](#)

Sélectionnez le CDN adapté à vos besoins

Le choix du bon fournisseur de réseau de diffusion de contenu (CDN) est une décision importante qui peut avoir un impact sur les performances de diffusion de contenu, les coûts et l'expérience des AWS Elemental MediaTailor utilisateurs. Tenez compte des facteurs suivants lors de la sélection d'un CDN pour votre MediaTailor implémentation :

Couverture géographique

Choisissez un CDN fortement présent dans les régions où se trouve votre public. Les différents fournisseurs de CDN ont des points forts différents selon les régions géographiques.

Intégration avec les AWS services

Amazon CloudFront propose l'intégration la plus étroite avec MediaTailor les autres AWS services, ce qui peut simplifier la configuration et la gestion. Les tiers CDNs peuvent offrir d'autres avantages, tels que des fonctionnalités de diffusion vidéo spécialisées ou une présence accrue dans des régions spécifiques.

Fonctionnalités spécifiques à la vidéo

Recherchez CDNs cette offre de fonctionnalités spécialement conçues pour la diffusion vidéo, telles que l'optimisation adaptative du débit, la compression vidéo et les analyses axées sur l'expérience du spectateur.

Structure des coûts

Comparez les modèles de tarification des différents fournisseurs, en tenant compte de facteurs tels que le volume de trafic, la distribution géographique et les exigences en matière de fonctionnalités. Certains CDNs proposent des remises sur volume ou des remises d'utilisation engagées qui peuvent correspondre à vos habitudes d'utilisation.

Support pour les fonctionnalités avancées

Vérifiez que le CDN que vous avez choisi prend en charge les fonctionnalités dont vous avez besoin, telles que l'authentification par jeton, la géo-restriction, le regroupement des demandes et le transfert d'en-tête approprié.

Pour plus d'informations sur les fournisseurs de CDN spécifiques et leur intégration avec ceux-ci MediaTailor, consultez les ressources suivantes.

- [intégration d'CloudFront](#) pour Amazon CloudFront
- [Configuration d'un CDN tiers](#) pour les fournisseurs de CDN tiers

Les rubriques suivantes fournissent des conseils complets sur la configuration à l' MediaTailor aide d'un CDN pour des performances optimales.

Planifiez votre intégration au CDN pour AWS Elemental MediaTailor

Vous pouvez améliorer l'expérience des spectateurs et réduire la latence grâce à une intégration CDN pour AWS Elemental MediaTailor. Lorsque vous implémentez un réseau de diffusion de contenu (CDN), vous pouvez diffuser du contenu depuis des emplacements plus proches de vos spectateurs. Cela garantit des temps de chargement plus rapides, une meilleure évolutivité et une diffusion publicitaire cohérente dans les différentes régions géographiques.

Vous avez besoin d'une bonne planification avant de mettre en œuvre un CDN avec AWS Elemental MediaTailor. Cette section vous guide à travers les principaux domaines de planification. Vous abordez ces domaines avant de commencer la configuration proprement dite. Ces étapes vous aident à créer une expérience de visionnage optimale pour votre public.

Une fois votre planification terminée, consultez la section [Intégration d'un CDN MediaTailor pour les instructions de](#) step-by-step mise en œuvre.

Pour plus d'informations sur MediaTailor les quotas susceptibles d'affecter la planification de votre CDN, consultez [Quotas en AWS Elemental MediaTailor](#). Pour plus d'informations sur CloudFront les quotas, consultez la section [Quotas](#) dans le guide du CloudFront développeur.

Avant de commencer à planifier, déterminez comment MediaTailor interagit avec un CDN :

1. Les spectateurs demandent du contenu via votre CDN plutôt que directement auprès de MediaTailor
2. Le CDN transmet les demandes de manifeste à des MediaTailor fins de personnalisation.
3. Le CDN met en cache et diffuse des segments de contenu et des segments publicitaires à partir d'emplacements périphériques.

Cette architecture réduit la charge de travail MediaTailor tout en garantissant que les spectateurs reçoivent des publicités personnalisées avec une latence minimale.

La compréhension de la terminologie du manifeste vous permet de configurer correctement votre CDN. Les différents protocoles de streaming utilisent des structures de manifeste spécifiques qui affectent la façon dont vous configurez la mise en cache et le routage :

- Manifestes HLS - Lorsque vous travaillez avec des flux HLS, vous gérez :

- Liste de lecture multivariée : configurez votre CDN pour qu'il achemine ces manifestes de haut niveau vers des fins de MediaTailor personnalisation.
- Liste de lecture multimédia : définissez des règles de mise en cache appropriées pour ces manifestes contenant des liens vers des segments de contenu.
- Manifestes DASH - Lorsque vous travaillez avec des flux DASH, vous gérez :
 - MPD (Media Presentation Description) : configurez votre CDN pour gérer ces manifestes conformément à vos exigences de personnalisation.

Le processus de planification du CDN comprend les étapes clés suivantes, chacune étant axée sur une tâche spécifique :

- [Estimation des besoins en trafic pour le CDN et les intégrations MediaTailor](#): calculez le nombre de spectateurs que vous attendez et les besoins en bande passante.
- [Configurer des stratégies d'optimisation pour le CDN et MediaTailor les intégrations](#): configurez votre CDN pour une diffusion de contenu et une personnalisation des publicités optimales.
- [Personnalisez la planification pour le CDN et les intégrations MediaTailor](#): ajustez votre stratégie CDN en fonction de votre MediaTailor flux de travail spécifique.
- [Configuration de la surveillance et de la mise à l'échelle pour le CDN et les intégrations MediaTailor](#): Mettez en œuvre des stratégies de surveillance et de mise à l'échelle pour des performances fiables.
- [Optimisez les coûts du CDN et MediaTailor des intégrations](#): Équilibrez les performances avec la rentabilité.
- [Testez votre implémentation pour le CDN et les intégrations MediaTailor](#): testez minutieusement votre intégration au CDN avant le déploiement en production.

Estimation des besoins en trafic pour le CDN et les intégrations MediaTailor

Pour dimensionner avec précision l'intégration de votre réseau de diffusion de contenu (CDN) avec AWS Elemental MediaTailor :

1. Calculez le nombre de spectateurs attendus à l'aide de données historiques ou d'événements similaires. Prévoyez une capacité supplémentaire au-delà de votre base de référence pour faire face à des pics inattendus. Pour connaître les recommandations de dimensionnement actuelles, consultez l'équipe chargée de votre AWS compte. Vous pouvez également consulter [Quotas en AWS Elemental MediaTailor](#).

2. Identifiez les modèles de trafic de pointe et les pics potentiels dans votre calendrier de contenu. Tenez compte de facteurs tels que :
 - Événements sportifs en direct ou avant-premières de saison
 - Campagnes marketing ou événements promotionnels
 - Différences de fuseaux horaires pour le public mondial
 - Schémas d'observation pendant les fêtes ou les saisons
3. Déterminez vos besoins en bande passante en multipliant le nombre de spectateurs par le débit des flux. Collaborez avec votre fournisseur de CDN pour calculer la capacité appropriée. Basez ce calcul sur vos débits de contenu spécifiques et sur la taille d'audience attendue. Ajoutez des frais supplémentaires pour les segments publicitaires et les demandes de manifeste, comme le recommande votre fournisseur.
4. Travaillez avec votre fournisseur de CDN pour garantir une capacité périphérique suffisante dans vos régions cibles.

Assurez-vous que votre capacité d'insertion publicitaire répond à la demande des internautes en prenant les mesures spécifiques suivantes :

1. Vérifiez votre quota actuel de demandes d'insertion d'annonces dans la [console Service Quotas](#). Passez en revue les limites de service actuelles pour savoir combien de spectateurs simultanés votre configuration peut prendre en charge.
2. Pour les événements à fort trafic, demandez un quota accru via la [console Service Quotas](#).
3. Si vous attendez plus de 500 000 spectateurs simultanés, contactez [AWS Support](#) au moins 2 semaines avant votre événement. Cela permet à AWS de garantir une capacité suffisante pour répondre à vos besoins de personnalisation des annonces.

Pour plus d'informations sur la mise en œuvre de la planification des capacités dans votre flux de travail, voir [Utilisation de la planification par prélecture pour](#) optimiser la diffusion des annonces lors d'événements à fort trafic.

Configurer des stratégies d'optimisation pour le CDN et MediaTailor les intégrations

Lorsque vous avez terminé l'estimation du trafic, configurez votre réseau de diffusion de contenu (CDN) pour optimiser la diffusion de contenu et la personnalisation des publicités avec AWS

Elemental MediaTailor Ces optimisations permettent de garantir une lecture fluide tout en maintenant une publicité ciblée.

Implémentez les optimisations spécifiques du CDN suivantes :

1. Configurez les fonctionnalités d'Origin Shield dans votre CDN afin de réduire la charge MediaTailor et d'améliorer l'efficacité de la mise en cache. Origin Shield agit comme une couche de mise en cache intermédiaire qui :

- Regroupez les demandes de plusieurs utilisateurs en une seule demande d'origine
- Réduisez le nombre de demandes redondantes pour MediaTailor
- Améliorez les temps de réponse pour le contenu mis en cache

Pour plus de détails sur la mise en œuvre de la configuration d'Origin Shield avec CloudFront, consultez la section [Utilisation d'Origin Shield](#) dans le manuel du CloudFront développeur.

2. Définissez les valeurs TTL (Time To Live) appropriées pour les différents types de contenu. Le TTL détermine la durée de mise en cache du contenu du CDN. Passé ce délai, le CDN demande une nouvelle copie à l'origine :

- Manifestes :
 - 0 secondes pour l'insertion de l'annonce
 - 5 à 10 secondes pour l'assemblage du canal

Lors de l'insertion d'annonces, MediaTailor fournit des manifestes contenant des publicités personnalisées pour le spectateur. Si une playlist ou un MPD est mis en cache et diffusé sur le mauvais appareil de lecture, celui-ci peut rencontrer des problèmes de lecture ou de suivi.

- Segments de contenu : 24 heures ou plus (ils changent rarement et vous pouvez les mettre en cache de manière agressive pour réduire la charge d'origine)
- Segments publicitaires : 24 heures ou plus (le contenu publicitaire est généralement réutilisé par les internautes et vous pouvez le mettre en cache pendant de longues périodes)

Pour des recommandations TTL complètes et des stratégies d'optimisation de la mise en cache pour tous les MediaTailor flux de travail, voir. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

Pour obtenir des instructions détaillées sur la configuration des comportements du cache dans CloudFront, consultez la section [Paramètres du comportement du cache](#) dans le Guide du CloudFront développeur.

3. Déployez des nœuds périphériques du CDN à proximité de vos populations de spectateurs.

Collaborez avec votre fournisseur de CDN pour :

- Identifiez les emplacements optimaux des nœuds périphériques en fonction de la démographie des spectateurs
- Garantir une capacité suffisante dans chaque région
- Surveillez les performances de pointe et ajustez-les selon les besoins

Pour obtenir des conseils de mise en œuvre, consultez la section [Emplacements CloudFront périphériques](#) afin d'identifier les emplacements périphériques disponibles pour les régions de votre audience.

4. Pour un public mondial, envisagez de mettre en œuvre une stratégie multi-CDN. Cette approche :

- Utilise plusieurs fournisseurs de CDN pour améliorer la fiabilité
- Oriente les spectateurs vers le CDN le plus performant pour leur localisation
- Fournit des options de basculement en cas de panne du CDN
- Peut optimiser les coûts en tirant parti de différents modèles de tarification

Pour plus de détails sur la mise en œuvre, consultez la section [Stratégies multi-CDN](#) sur le blog AWS Networking & Content Delivery.

Personnalisez la planification pour le CDN et les intégrations MediaTailor

Les différents AWS Elemental MediaTailor flux de travail ont des exigences uniques qui affectent la planification du réseau de diffusion de contenu (CDN). Pour obtenir des conseils spécifiques au flux de travail, consultez la section [Utilisation](#) des configurations.

Ajustez votre plan de capacité en fonction de votre MediaTailor flux de travail spécifique :

Pour les flux de travail d'insertion de MediaTailor publicités

1. Configurez votre CDN pour gérer des manifestes personnalisés sans aucune mise en cache. Cela garantit que chaque spectateur reçoit des publicités uniques et ciblées. Les publicités sont basées sur leur profil et leur contexte de visionnage.
2. Dimensionnez votre serveur de décision publicitaire (ADS) pour gérer les pics de demandes. Pour obtenir des conseils sur la configuration d'ADS, consultez la section [Insertion d'annonces avec MediaTailor](#). Tenez compte de :
 - Temps de réponse requis pour votre cas d'utilisation

- Capacité de visionnage simultanée attendue
 - Exigences en matière de redondance et de basculement
 - Besoins de distribution géographique
3. Implémentez le regroupement des demandes au niveau du CDN pour gérer les demandes d'interruption de publicité synchronisées. Le regroupement des demandes combine plusieurs demandes identiques en une seule demande d'origine. Cela est crucial lors de :
- Événements sportifs en direct lorsque de nombreux spectateurs font des pauses publicitaires simultanément
 - Émissions de télévision populaires diffusées en avant-première avec des pauses publicitaires synchronisées
 - Des événements d'actualité qui attirent des pics soudains d'audience

Pour plus de détails sur la mise en œuvre, consultez [Origin Failover](#) pour configurer le traitement des demandes pendant les pics de charge.

Pour les flux de travail d'assemblage de MediaTailor canaux

1. Calculez les besoins en capacité en fonction du nombre de canaux et de leurs débits. Pour obtenir des conseils sur la planification de la capacité d'assemblage des canaux, voir [Assemblage des canaux dans MediaTailor](#). Tenez compte de :
- Nombre total de canaux
 - Exigences en matière de débit par canal
 - Charge de visionnage simultanée attendue
 - Besoins de distribution géographique
2. Configurez votre CDN pour gérer des modèles de trafic prévisibles en fonction des plannings publiés. L'assemblage des chaînes présente généralement des modèles plus prévisibles que l'insertion de publicités pour les raisons suivantes :
- Les horaires de programmation sont connus à l'avance
 - Le comportement du spectateur suit des modèles établis
 - Le contenu ne change pas de manière dynamique par spectateur
3. Assurez-vous que votre origine dispose d'une bande passante suffisante pour maintenir une sortie de canal cohérente. Mettre en œuvre :
- [Serveurs d'origine redondants pour des canaux à haute disponibilité](#)

- Basculement automatique entre l'origine principale et l'origine de sauvegarde
- Surveillance pour détecter les problèmes de performance liés à l'origine

Pour obtenir des conseils de mise en œuvre, voir [Configuration de la redondance d'origine](#) pour créer une infrastructure d'origine résiliente.

Pour des MediaTailor flux de travail combinés

1. Dimensionnez votre infrastructure pour gérer les modèles de trafic combinés des deux services. Pour obtenir des conseils sur les flux de travail combinés, voir [Utilisation AWS Elemental MediaTailor pour insérer des publicités](#). Tenez compte de :
 - Exigences de base relatives à l'assemblage des canaux
 - Frais généraux d'insertion d'annonces
 - Modèles de trafic de pointe
 - Besoins en matière de redondance
2. Configurez des comportements CDN distincts pour la diffusion de contenu linéaire et l'insertion dynamique de publicités. Cette séparation vous permet de :
 - Optimisez les politiques de mise en cache pour chaque type de contenu indépendamment
 - Acheminer les demandes vers les origines appropriées en fonction du type de contenu
 - Surveillez les indicateurs de performance séparément pour chaque flux de travail
3. Configurez un routage approprié entre la périphérie et l'origine CDN pour maintenir des performances optimales. Pensez à utiliser :
 - Différents chemins d'origine pour les segments de contenu (/content/*) et les segments publicitaires (/ads/*)
 - Comportements de cache distincts pour les manifestes et les segments
 - Routage géographique pour optimiser le temps de latence pour les différentes régions

Pour plus de détails sur l'implémentation, consultez [la section Configuration des comportements du cache](#) pour configurer des règles de routage et de mise en cache basées sur le chemin.

Configuration de la surveillance et de la mise à l'échelle pour le CDN et les intégrations MediaTailor

Des stratégies de surveillance et de mise à l'échelle efficaces sont essentielles pour maintenir des performances et une expérience utilisateur optimales grâce à l'intégration de votre réseau de diffusion de AWS Elemental MediaTailor contenu (CDN). Mettez en œuvre ces approches pour garantir que votre intégration CDN fonctionne de manière fiable à grande échelle.

Mettez en œuvre les stratégies de surveillance et de mise à l'échelle suivantes :

1. Configurez la surveillance pour ces indicateurs clés. Pour obtenir des conseils sur les valeurs cibles appropriées, consultez [Monitoring MediaTailor with Amazon CloudWatch](#) et consultez les meilleures pratiques de votre fournisseur de CDN :
 - Taux de réussite du cache CDN (établisseez des indicateurs et des cibles de référence en fonction de votre type de contenu et de vos habitudes de diffusion)
 - Volumes de demandes d'origine (surveillez les modèles pendant le fonctionnement normal afin d'établir des bases de référence pour la détection des anomalies)
 - Taux d'erreur par type d'erreur (définissez des seuils en fonction de vos objectifs de niveau de service et des MediaTailor meilleures pratiques)
 - Temps de réponse (définissez des objectifs de latence appropriés en fonction de vos exigences en matière d'expérience utilisateur et de votre répartition géographique)

Pour obtenir des instructions de mise en œuvre détaillées, consultez [la section Création de CloudWatch tableaux](#) de bord pour visualiser vos indicateurs MediaTailor et ceux du CDN ensemble.

2. Configurez des alertes en cas de modèles de trafic inattendus ou de dégradation des performances. Configurez les seuils en fonction de vos indicateurs de référence et de vos objectifs de niveau de service. Pour obtenir des conseils sur la configuration des alertes, consultez la section [Création d' CloudWatch alarmes Amazon](#). Envisagez de surveiller :
 - Écarts importants par rapport aux taux d'accès au cache de référence (alerte généralement en cas de baisse de 85 à 90 %)
 - Augmentations soudaines du volume de demandes d'origine (alerte en cas d'augmentation de 30 % ou plus par rapport à la valeur de référence)
 - Les pics de taux d'erreur dépassent les seuils que vous avez définis (généralement 1 à 2 % pour les erreurs 4xx, 0,5 % pour les erreurs 5xx)

- Dégradation du temps de réponse au-delà des niveaux acceptables (généralement > 500 ms pour les manifestes, > 200 ms pour les segments)

Pour des exemples de mise en œuvre, consultez [CloudWatch les concepts](#) de création de tableaux de bord et d'alertes de surveillance efficaces.

3. Créez des plans de mise à l'échelle pour les événements prévisibles à fort trafic. Vos plans doivent inclure les éléments clés suivants :

- Augmentations de capacité avant l'événement (24 à 48 heures avant le début de l'événement)
- Horaires d'augmentation progressive du nombre de spectateurs (généralement 10 à 20 % de l'audience attendue par intervalle de 5 minutes)
- Répartition des capacités régionales en fonction de l'audience (attribuer la capacité proportionnellement à l'audience régionale attendue)
- Procédures de dimensionnement après l'événement (maintien de la capacité maximale pendant 30 à 60 minutes après la fin de l'événement)

Pour obtenir des conseils de mise en œuvre sur la mise à l'échelle en cas d'événements à fort trafic, consultez la section [Configuration d'un flux de travail end-to-end en direct résilient](#) sur le blog AWS des médias.

4. Mettez en œuvre des mesures de basculement et de redondance pour les flux critiques, notamment :

- Déploiements de CDN multirégionaux (au moins deux régions pour le contenu critique)
- Serveurs d'origine de sauvegarde (configurés avec des contrôles de santé automatisés toutes les 30 secondes)
- Déclenchements de basculement automatisés basés sur des tests de santé (généralement après 2 ou 3 vérifications infructueuses)
- Procédures de restauration pour différents scénarios de défaillance (documentées avec des objectifs de temps de réponse spécifiques)

Pour connaître les étapes de mise en œuvre détaillées, voir [Optimisation de la haute disponibilité avec le basculement CloudFront d'origine](#).

Optimisez les coûts du CDN et MediaTailor des intégrations

Les coûts des réseaux de diffusion de contenu (CDN) peuvent varier considérablement en fonction des modèles de trafic, de la distribution géographique et de l'utilisation des AWS Elemental

MediaTailor fonctionnalités. Pour obtenir des informations sur les prix actuels, consultez la section [CloudFront sur les prix](#). Vous pouvez également consulter la documentation de votre fournisseur de CDN.

Équilibrez les performances avec le rapport coût-efficacité en utilisant les stratégies suivantes :

1. Analysez les modèles de trafic de votre CDN pour sélectionner le niveau de tarification le plus rentable. Passez en revue les facteurs suivants avec votre fournisseur de CDN :

- Volumes de transfert de données par région et par période
- Modèles de demande pour les manifestes et les segments
- Exigences relatives à la distribution géographique
- Modes d'utilisation de pointe par rapport à la moyenne

Pour vous aider à analyser les coûts, utilisez le [calculateur de AWS prix](#) pour estimer vos CloudFront coûts en fonction de vos habitudes d'utilisation spécifiques.

2. Pour des charges de travail prévisibles, évaluez les accords de capacité réservée avec votre fournisseur de CDN. Ces accords peuvent offrir des avantages tels que :

- Tarifs réduits pour les volumes d'utilisation engagés
- Coûts mensuels prévisibles pour la budgétisation
- Soutien prioritaire et allocation de capacités

Consultez votre fournisseur de CDN pour déterminer si la capacité réservée est adaptée à vos habitudes d'utilisation. Pour CloudFront, voir les [fonctionnalités CloudFront premium](#). Cela fournit des informations sur les options de capacité réservée.

3. Optimisez les coûts de sortie en équilibrant le trafic entre MediaTailor et votre fournisseur de CDN. Les stratégies incluent :

- Maximisez les taux de réussite du cache pour réduire les demandes d'origine
- Utiliser Origin Shield pour consolider les demandes
- Implémenter la compression pour réduire les volumes de transfert de données
- Choisissez les régions du CDN qui correspondent à AWS la tarification régionale

Pour obtenir des conseils d'implémentation sur la compression, consultez la section [Servir des fichiers compressés](#) dans le manuel du CloudFront développeur.

4. Mettez en œuvre des stratégies de mise en cache appropriées pour les différents types de contenu afin de réduire les demandes d'origine. Pour obtenir des conseils sur l'optimisation du cache, consultez [la section Amélioration des taux de réussite du cache](#). Concentrez-vous sur :

- Segments de contenu (vous pouvez les mettre en cache pendant de longues périodes)
- Segments publicitaires (généralement réutilisés par les utilisateurs)
- Ressources statiques telles que les fichiers de lecteur et les images

L'amélioration des taux d'accès au cache réduit considérablement les coûts d'origine. Collaborez avec votre fournisseur de CDN pour optimiser les configurations de cache en fonction de vos modèles de contenu spécifiques. Pour connaître les étapes de mise en œuvre détaillées, consultez [la section Configuration des comportements du cache](#).

Testez votre implémentation pour le CDN et les intégrations MediaTailor

Garantissez une expérience utilisateur fiable en testant minutieusement l'intégration de votre réseau de diffusion de AWS Elemental MediaTailor contenu (CDN) avant le déploiement en production. Des tests appropriés permettent d'identifier et de résoudre les problèmes avant qu'ils n'affectent votre public. Pour obtenir des conseils sur les méthodologies de test, voir [Tester CloudFront les distributions](#). Vous pouvez également consulter la documentation de test de votre fournisseur de CDN.

Suivez ces étapes de test pour valider votre intégration au CDN :

1. Créez un environnement de test qui reflète votre configuration de production. Inclure :
 - Paramètres CDN et comportements de cache identiques
 - Incluez du contenu représentatif avec différents débits et formats
 - Configuration d'un serveur de test et de décision publicitaire avec des exemples de réponses publicitaires
 - Configuration des configurations de surveillance et d'alerte

Pour obtenir des conseils de step-by-step mise en œuvre, consultez [la section Création d'une distribution intermédiaire](#) dans le Guide du CloudFront développeur.

2. Effectuez des tests de charge pour vérifier vos estimations de capacité. Pour obtenir des conseils sur les tests de charge, consultez [la section Surveillance MediaPackage](#). Les scénarios de test doivent inclure :

- Augmentation progressive du nombre de spectateurs (généralement 10 à 20 % du pic attendu toutes les 5 minutes)
- Des pics de trafic soudains en fonction de vos pics de charge attendus (simulez 50 % du pic d'audience rejoignant le site dans les 60 secondes)
- Périodes de pointe prolongées (maintien de la charge maximale pendant au moins 30 à 60 minutes)
- Distribution géographique correspondant à votre audience (répartissez le trafic de test en fonction de la localisation attendue des spectateurs)

Vérifiez que les temps de réponse restent inférieurs aux seuils cibles. Cela signifie généralement moins de 500 ms pour les manifestes et moins de 200 ms pour les segments. Le taux d'erreur doit rester inférieur à 1 %. Pour plus de détails sur la mise en œuvre des outils et méthodologies de test de [charge](#), consultez la [section relative aux tests](#) de charge CloudFront sur le blog AWS Networking & Content Delivery.

3. Testez des scénarios de basculement pour garantir la fiabilité. Simulez :

- Défaillances du serveur d'origine (scénarios de panne complète et de dégradation partielle)
- Interruptions des sites périphériques du CDN (test avec routage du trafic vers des sites de sauvegarde)
- Indisponibilité du serveur Ad Decision (test avec des délais d'expiration de 5 à 10 secondes)
- Problèmes de connectivité réseau (simulation de la perte de paquets et de l'augmentation de la latence)

Travaillez avec votre fournisseur de CDN pour établir des objectifs de temps de réponse en cas de basculement adaptés à votre cas d'utilisation. Généralement, cela signifie moins de 3 secondes pour terminer le basculement. Pour obtenir des conseils d'implémentation sur les tests de basculement, voir [Origin failover](#) dans le guide du CloudFront développeur.

4. Pour les événements majeurs, mettez en œuvre des stratégies de montée en puissance progressive basées sur les directives relatives aux [tests AWS de charge](#) :

- Étalez les heures de début en fonction des différents segments d'audience. Par exemple, autorisez d'abord les abonnés premium avec des intervalles de 15 minutes entre les groupes d'audience.
- Utilisez des techniques de préchauffage pour augmenter progressivement la charge. Le préchauffage implique :
 - Remplir les caches du CDN avec du contenu populaire 2 à 4 heures avant l'événement

- Augmenter progressivement le trafic synthétique jusqu'à 20 à 30 % du pic prévu pour réchauffer les systèmes
- Testez tous les composants dans des conditions de charge réalistes avec le contenu réel
- Surveillez les performances du système tout au long de la période de mise en service, en suivant :
 - Taux d'accès au cache et temps de réponse (objectif : taux de réussite supérieur à 90 %, réponse inférieure à 500 ms)
 - Taux d'erreur et charge d'origine (maintien des taux d'erreur inférieurs à 1 %, processeur d'origine < 70 %)
 - Taux de réussite de la personnalisation des publicités (objectif > 98 % de personnalisation réussie)
 - Mesures relatives à l'expérience utilisateur (délai de démarrage cible inférieur à 2 secondes, délai de rebufferisation inférieur à 0,5 %)
- Établissez un plan d'urgence en cas de surtension imprévue. Votre plan doit inclure les éléments essentiels suivants :
 - Procédures de mise à l'échelle des capacités d'urgence avec étapes documentées pour augmenter la capacité de 50 à 100 % en 15 minutes
 - Protocoles d'activation du CDN de sauvegarde avec possibilité de transférer 20 à 50 % du trafic vers le CDN secondaire
 - Solution de repli simplifiée pour l'insertion d'annonces afin de réduire les paramètres de ciblage de 10 ou plus à 3 à 5 paramètres essentiels
 - Plans de communication pour les parties prenantes avec des modèles de notification et des listes de contacts prédéfinis

Une fois vos tests terminés, passez à la section [Implémentation de votre intégration CDN](#) pour les étapes de déploiement en production.

Résoudre les problèmes courants liés au CDN et aux intégrations MediaTailor

Relevez les problèmes courants d'intégration des réseaux de diffusion de contenu (CDN) AWS Elemental MediaTailor avant qu'ils n'aient un impact sur vos spectateurs. Cette section vous aide à identifier et à résoudre les problèmes typiques qui se produisent lors de l'intégration du CDN avec [AWS Elemental MediaTailor](#).

Pour obtenir des conseils de dépannage complets, consultez la section [Résolution des problèmes MediaTailor](#) et [CloudFront distributions de résolution](#) des problèmes.

Résolution des problèmes de livraison de MediaTailor manifestes

Si les spectateurs rencontrent des problèmes de lecture ou voient des publicités incorrectes, vérifiez les problèmes courants liés aux manifestes suivants :

- Paramètres de mise en cache incorrects : si votre CDN met en cache des manifestes personnalisés, les spectateurs peuvent voir des publicités destinées à d'autres utilisateurs.

Solution : configurez votre CDN avec un TTL de cache de 0 pour les demandes de manifeste adressées à MediaTailor

- Échec des demandes d'origine : si votre CDN ne parvient pas à atteindre MediaTailor, les demandes de manifeste échoueront.

Solution : Vérifiez la connectivité réseau entre votre CDN et MediaTailor. Vérifiez que votre CDN transmet correctement les en-têtes corrects.

- Problèmes liés aux paramètres de session : des paramètres de session manquants ou incorrects peuvent entraîner des échecs de personnalisation.

Solution : assurez-vous que votre joueur ajoute correctement tous les paramètres de session requis aux demandes de manifeste.

Résoudre les problèmes de diffusion des MediaTailor segments

Si le contenu ou les segments publicitaires ne se chargent pas correctement, examinez les problèmes courants suivants :

- Réécriture du chemin du segment : une configuration CDN incorrecte peut réécrire le segment de manière incorrecte. URLs

Solution : Vérifiez que votre CDN gère correctement le segment URLs. Assurez-vous qu'il ne modifie pas les chemins d'une manière qui rompe les références.

- Configuration CORS : des en-têtes CORS manquants ou incorrects peuvent empêcher les navigateurs de charger des segments.

Solution : configurez votre CDN pour transmettre les en-têtes CORS appropriés aux demandes de segment.

- Tempêtes d'erreurs de cache : lors d'événements à fort trafic, plusieurs défaillances de cache peuvent submerger les serveurs d'origine.

Solution : implémentez des fonctionnalités de réduction des demandes et de protection des origines afin de réduire la charge d'origine lors des pics de trafic.

Résolution des problèmes de performance MediaTailor du CDN

Si les spectateurs sont confrontés à une mise en mémoire tampon ou à un chargement lent, vérifiez les problèmes de performances suivants :

- Faible taux de réussite du cache : si votre CDN demande fréquemment du contenu depuis l'origine, les performances en pâtiront.

Solution : Analysez les taux d'accès au cache par type de contenu et ajustez les paramètres TTL pour améliorer l'efficacité de la mise en cache.

- Distribution géographique : les spectateurs éloignés des emplacements périphériques du CDN peuvent être confrontés à une latence accrue.

Solution : passez en revue la distribution des emplacements périphériques de votre CDN. Augmentez la capacité dans les régions à forte concentration de spectateurs.

- Limitations de capacité d'origine : si vos serveurs d'origine sont surchargés, les temps de réponse augmenteront.

Solution : implémentez la limitation des demandes d'origine. Vous pouvez également augmenter la capacité d'origine ou améliorer la mise en cache pour réduire la charge d'origine.

Pour obtenir une assistance supplémentaire en matière de dépannage, consultez la section [Dépannage MediaTailor](#) ou contactez AWS le Support. Pour obtenir des conseils de mise en œuvre sur la résolution des problèmes courants liés au CDN, consultez la section [Débogage de votre réseau de diffusion de contenu](#) sur le blog des AWS médias.

Configurer l'intégration du CDN avec MediaTailor

Cette section fournit des conseils sur l'intégration AWS Elemental MediaTailor à un réseau de diffusion de contenu (CDN).

L'intégration efficace du CDN MediaTailor est essentielle pour proposer des expériences de streaming de haute qualité avec des publicités personnalisées à grande échelle. Ce guide explique le processus complet de configuration, de configuration et d'optimisation de votre intégration au CDN.

Pour plus d'informations, consultez les liens suivants :

- Pour plus d'informations sur la transmission de paramètres de requête à CDNs des fins d'autorisation et de ciblage publicitaire, consultez [MediaTailor paramètres de requête manifeste](#).
- Pour un routage avancé utilisant des variables dynamiques et des alias de configuration, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).
- Pour plus d'informations sur la création de MediaTailor configurations, consultez [Utilisation AWS Elemental MediaTailor pour insérer des publicités](#).
- Pour plus d'informations sur la création d'une CloudFront distribution, consultez [la section Création d'une distribution](#) dans le Guide du CloudFront développeur.

Pour garantir la clarté de ce guide, la terminologie suivante est utilisée pour décrire les différents types de manifestes :

- Manifestes HLS :
 - Liste de lecture multivariée : manifeste de niveau supérieur contenant des liens vers des listes de lecture multimédia
 - Liste de lecture multimédia : le manifeste de deuxième niveau contenant des liens vers des segments de contenu
- DASH manifeste :
 - MPD (Media Presentation Description) : terme standard pour les manifestes DASH

L'intégration MediaTailor à un CDN offre les avantages suivants :

- Latence réduite pour les spectateurs
- Évolutivité améliorée pour les événements à fort trafic
- Fiabilité accrue grâce à des voies de livraison redondantes
- Coûts optimisés en réduisant le trafic d'origine
- Meilleure protection contre les attaques DDoS

Composants et exigences d'intégration du CDN

Une intégration CDN réussie MediaTailor implique de configurer les composants clés suivants :

Comportements de routage du CDN

Règles qui déterminent la manière dont les différents types de demandes (manifestes, segments de contenu, segments publicitaires) sont acheminés via votre CDN.

Mappage du CDN dans MediaTailor

Configuration garantissant MediaTailor que les manifestes font référence à votre domaine CDN au lieu de référencer directement les serveurs d'origine.

Paramètres de sécurité

Des configurations qui protègent votre contenu et votre infrastructure, notamment en matière de sécurité du transport, de contrôle d'accès et de surveillance.

Tests et validation

Procédures permettant de vérifier que votre intégration CDN fonctionne correctement avant le déploiement en production.

Conditions préalables à l'intégration du CDN

Avant de configurer votre intégration au CDN, assurez-vous de disposer des éléments suivants :

1. Une MediaTailor configuration avec les paramètres suivants :

- L'origine de votre contenu en tant que source de contenu
- Votre ADS en tant que serveur de décision publicitaire

Vous avez également besoin de l'origine et de l'ADS URLs dans les étapes d'intégration du CDN.

2. Accès à l'interface de configuration de votre CDN

3. Compréhension de la terminologie spécifique de votre CDN pour les comportements, les règles et les paramètres de cache

4. Connaissance de la structure de votre contenu, y compris des extensions de fichier utilisées pour les segments (telles que .ts, .mp4 ou .m4s)

Étapes de configuration de l'intégration du CDN

Le processus d'intégration MediaTailor à un CDN suit les étapes de haut niveau suivantes :

1. Configurer les comportements de routage du CDN : configurez votre CDN pour acheminer les différents types de demandes de manière appropriée.
2. Configurez le mappage du CDN dans MediaTailor - Mettez à jour votre MediaTailor configuration pour utiliser vos noms de domaine CDN.
3. Mettez en œuvre les meilleures pratiques de sécurité : configurez les paramètres de sécurité pour protéger votre contenu et votre infrastructure.
4. Testez votre intégration : vérifiez que l'intégration de votre CDN fonctionne correctement.

En-têtes requis pour l'intégration du MediaTailor CDN

MediaTailor Pour fonctionner correctement avec votre CDN, vous devez configurer votre CDN pour transférer des en-têtes HTTP spécifiques. Ces en-têtes sont essentiels au bon fonctionnement, notamment pour la compression, la détection des appareils, la personnalisation des publicités et le ciblage géographique.

Configurez votre CDN pour transférer les en-têtes suivants vers : MediaTailor

Accept-Encoding

Objectif : Nécessaire pour la fonctionnalité de compression

Détails : Cet en-tête indique MediaTailor les méthodes de compression prises en charge par le client. MediaTailor utilise ces informations pour compresser les manifestes lorsque cela est possible, afin de réduire l'utilisation de la bande passante et d'améliorer les performances. Les anciens appareils qui ne prennent pas en charge la compression n'envoient pas cet en-tête et MediaTailor renverra des manifestes non compressés.

User-Agent

Objectif : Nécessaire pour la détection des appareils et la personnalisation des publicités

Détails : MediaTailor analyse l'en-tête User-Agent pour identifier le type d'appareil client, le navigateur et les fonctionnalités. Ces informations sont utilisées pour le ciblage publicitaire, les optimisations spécifiques à l'appareil et pour garantir la compatibilité avec les différents clients de lecture.

Host

Objectif : Nécessaire pour le bon acheminement des demandes

Détails : L'Host en-tête garantit que les demandes sont acheminées vers le point de MediaTailor terminaison approprié. Cela est particulièrement important dans les environnements à locataires multiples et lors de l'utilisation de configurations de domaine personnalisées.

Beaucoup CDNs, y compris Amazon CloudFront, ne transmettent pas l'Host en-tête par défaut. Pour CloudFront les utilisateurs : voir [Configuration des comportements du cache](#) les instructions de configuration.

X-Forwarded-For

Objectif : Nécessaire pour la détection de l'adresse IP des clients et le ciblage géographique

Détails : Cet en-tête préserve l'adresse IP d'origine du client lorsque les demandes transitent par votre CDN. MediaTailor utilise ces informations à des fins de ciblage publicitaire géographique, d'analyse et de conformité aux restrictions de contenu régionales.

Important

Les quatre en-têtes sont nécessaires pour une MediaTailor fonctionnalité complète. L'absence de l'un de ces en-têtes peut entraîner une réduction des fonctionnalités, notamment :

- Impossible de compresser les manifestes (Accept-Encoding manquant)
- Mauvais ciblage publicitaire et problèmes de compatibilité des appareils (agent utilisateur manquant)
- Défaillances du routage des demandes (hôte manquant)
- Ciblage géographique et analyses inexacts (manquants) X-Forwarded-For

Pour obtenir des instructions de configuration spécifiques au CDN, consultez les sections sur les comportements de routage et la mise en cache qui font référence à cette liste d'en-têtes.

Les rubriques suivantes fournissent des instructions détaillées pour chaque aspect de l'intégration du CDN.

Rubriques

- [Configurer les comportements de routage du CDN pour MediaTailor](#)
- [Configurer le mappage du CDN dans MediaTailor](#)
- [Bonnes pratiques de sécurité de l'intégration CDN pour MediaTailor](#)

Configurer les comportements de routage du CDN pour MediaTailor

Cette section explique comment configurer votre réseau de diffusion de contenu (CDN) pour acheminer différents types de demandes de manière appropriée à des fins d' AWS Elemental MediaTailor intégration. Une configuration de routage appropriée garantit que les demandes de manifeste, les segments de contenu et les segments publicitaires sont traités correctement.

La configuration des comportements de routage des CDN est une étape essentielle pour créer un pipeline de diffusion de contenu efficace. En définissant des règles de routage spécifiques pour différents types de contenu, vous pouvez optimiser la mise en cache, améliorer les performances de diffusion et garantir le bon fonctionnement de l'insertion d'annonces personnalisées.

Pour les scénarios de routage avancés utilisant des variables dynamiques et des alias de configuration, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour plus d'informations sur la préservation des paramètres de requête sur le routage du CDN, consultez [MediaTailor paramètres de requête manifeste](#).

Important

Le fait de ne pas inclure les en-têtes CORS dans l'objet mis en cache ou dans la réponse du CDN aux spectateurs peut entraîner des échecs de lecture.

Configuration du comportement de routage du CDN

Configurez votre CDN pour acheminer les différents types de demandes de manière appropriée.

Routage des segments de contenu

Le routage des segments de contenu dirige les demandes relatives à vos segments de contenu réels vers votre serveur d'origine. Tout comme le routage des segments publicitaires, le routage des segments de contenu nécessite également une configuration CORS appropriée pour garantir une lecture fluide sur les lecteurs Web.

Pour obtenir des conseils de configuration détaillés, consultez l' CloudFront exemple sur [Priorité 4 : Comportement du chemin d'origine du contenu](#). Cet exemple fournit des paramètres spécifiques que vous devez suivre CloudFront ou adapter pour d'autres CDNs.

Les principales exigences de configuration pour le routage des segments de contenu sont les suivantes :

- Utilisez des modèles de chemin qui correspondent aux extensions de fichiers de votre segment de contenu (comme *.ts*.mp4, ou *.m4s)
- Acheminez les demandes vers l'origine de votre contenu (tel qu'un compartiment ou un point de MediaPackage terminaison Amazon S3)
- Pour un ratio cache-accès optimal, incluez uniquement les paramètres de chaîne de requête qui obligent votre origine à modifier la réponse dans la clé de cache et à transférer la demande d'origine
- Appliquez une politique de cache appropriée avec des valeurs TTL supérieures à 24 heures
- Incluez les en-têtes de réponse CORS à vos spectateurs

Routage des segments publicitaires

Le routage des segments publicitaires est essentiel pour proposer des publicités personnalisées aux spectateurs. Lorsque vous configurez le routage des segments publicitaires, vous devez implémenter une gestion CORS (Cross-Origin Resource Sharing) appropriée afin d'éviter les problèmes susceptibles d'entraîner des échecs de lecture sur les lecteurs Web.

Pour obtenir des conseils de configuration détaillés, consultez l' CloudFront exemple sur [Priorité 0 : comportement du chemin des segments publicitaires](#). Cet exemple fournit des paramètres spécifiques que vous devez suivre CloudFront ou adapter pour d'autres CDNs.

Les principales exigences de configuration pour le routage des segments publicitaires sont les suivantes :

- Utilisez le modèle de chemin /tm/* spécifiquement pour les segments MediaTailor publicitaires
- Acheminer les demandes vers segments.mediatailor.*region*.amazonaws.com
- Pour un ratio cache/accès optimal, n'incluez aucun en-tête de demande d'affichage, aucun cookie ou paramètre de chaîne de requête dans la clé de cache ou dans la demande d'origine
- Appliquez une politique de cache appropriée avec des valeurs TTL supérieures à 24 heures
- Incluez les en-têtes de réponse CORS à vos spectateurs

Routage des demandes de manifeste

Pour acheminer les demandes de liste de lecture multivariante, de liste de lecture multimédia et de MPD vers MediaTailor, utilisez les paramètres généraux suivants. Pour CloudFront la configuration, voir [Configuration du comportement du cache du manifeste](#).

1. Dans votre interface de configuration CDN, créez des comportements pour différents types de manifestes.
2. Définissez des modèles de chemin correspondant aux extensions de fichiers de playlist et de playlist multimédia multivariantes (* .m3u8 pour HLS) et aux extensions de fichier MPD (* .mpd pour DASH).
3. Configurez le paramètre d'origine dans votre CDN pour qu'il pointe vers votre point de terminaison MediaTailor de configuration.
4. Pour l'insertion d'annonces, désactivez la mise en cache des playlists multivariantes personnalisées, des playlists multimédias et. MPDs Étant donné que l'insertion d'annonces fournit des manifestes personnalisés, votre CDN ne doit pas les mettre en cache. Si un appareil de lecture différent de celui prévu reçoit une playlist ou un MPD mis en cache, cela peut entraîner des problèmes de lecture ou de suivi. Pour obtenir des conseils complets sur la mise en cache, y compris des recommandations TTL pour tous les types de contenu, consultez. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
5. Configurez le transfert d'en-têtes pour tous les en-têtes. Pour les exigences minimales, voir [En-têtes requis pour l'intégration du MediaTailor CDN](#).
6. Activez le transfert de chaînes de requêtes pour transmettre les paramètres de ciblage publicitaire.

HLS multivariant playlist

Les demandes de playlist multivariantes HLS suivent les formats suivants :

```
https://<playback-endpoint>/v1/index/<hashed-account-id>/<origin-id>/<index>.m3u8
```

Exemple :

```
https://777788889999.mediataylor.us-east-1.amazonaws.com/v1/master/  
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/Demo/assetId.m3u8
```

HLS media playlist

Les demandes de playlist multimédia HLS suivent les formats suivants :

```
https://<playback-endpoint>/v1/manifest/<hashed-account-id>/<session-id>/  
<manifestNumber>.m3u8
```

Les demandes du joueur `https://CDN_Hostname/some/path/asset.m3u8` sont acheminées vers le MediaTailor chemin en `https://mediatailor.us-west-2.amazonaws.com/v1/session/configuration/endpoint` fonction du mot clé contenu `*.m3u8` dans la demande.

Exemple :

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/manifest/  
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/c240ea66-9b07-4770-8ef9-7d16d916b407/0.m3u8
```

DASH MPD

Les demandes DASH MPD suivent les formats suivants :

```
https://<playback-endpoint>/v1/dash/<hashed-account-id>/<origin-id>/<assetName>.mpd
```

Les demandes du joueur `https://CDN_Hostname/some/path/asset.mpd` sont acheminées vers le MediaTailor chemin en `https://mediatailor.us-west-2.amazonaws.com/v1/dash/configuration/endpoint` fonction du mot clé contenu `*.mpd` dans la demande.

Exemple :

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/  
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/Demo/0.mpd
```

Bonnes pratiques de routage CDN

Lorsque vous configurez les comportements de routage du CDN, suivez les meilleures pratiques suivantes pour garantir des performances et une fiabilité optimales :

Utiliser des modèles de trajectoire spécifiques

Créez des modèles de chemin spécifiques qui correspondent exactement à la structure de votre contenu afin de garantir un routage correct.

Prioriser l'ordre des comportements

Dans la plupart des CDN cas, les comportements sont évalués dans l'ordre. Placez les comportements plus spécifiques avant les comportements plus généraux.

Modèles de comportement de test

Vérifiez que vos modèles de chemin correspondent correctement aux demandes attendues avant le déploiement en production.

Documentez votre configuration

Conservez la documentation de vos comportements de routage CDN pour faciliter le dépannage et les futures mises à jour.

Étapes suivantes

Après avoir configuré les comportements de routage de votre CDN, l'étape suivante consiste à configurer le mappage du CDN dans MediaTailor. Pour obtenir des instructions, consultez [Configurer le mappage du CDN dans MediaTailor](#).

Configurer le mappage du CDN dans MediaTailor

Cette section explique comment configurer l'utilisation des noms AWS Elemental MediaTailor de domaine de votre réseau de diffusion de contenu (CDN). Après avoir configuré vos comportements de routage CDN, vous devez mettre à jour votre MediaTailor configuration pour vous assurer que les manifestes font référence à votre domaine CDN au lieu de faire directement référence aux serveurs d'origine.

La configuration du mappage du CDN MediaTailor garantit que tous les contenus et segments URLs publicitaires de vos manifestes pointent vers votre CDN plutôt que directement vers les serveurs d'origine. Cette étape est essentielle pour créer une chaîne de distribution CDN complète et optimiser les avantages de votre intégration CDN.

Configuration du mappage CDN dans MediaTailor

Après avoir configuré vos comportements de routage CDN, configurez MediaTailor pour utiliser vos noms de domaine CDN :

1. Ouvrez la [MediaTailor console](#).

2. Sélectionnez la configuration que vous souhaitez mettre à jour.
3. Dans les paramètres avancés de la section Configuration du CDN, entrez votre nom de domaine CDN dans le champ du préfixe du segment de contenu CDN.
4. Si vous utilisez un domaine CDN distinct pour les segments publicitaires, saisissez-le dans le champ de préfixe du segment publicitaire CDN.
5. Enregistrez vos modifications.

Cette configuration garantit la génération MediaTailor de manifestes pointant vers votre CDN plutôt qu'URLs que directement vers le stockage d'origine ou de segment publicitaire.

Comprendre le comportement des URL de base

MediaTailor détermine la base URLs dans les manifestes en fonction de la configuration de votre préfixe CDN :

- Préfixe de segment publicitaire CDN configuré : les segments publicitaires utilisent le préfixe CDN comme URL de base.
- Le préfixe du segment publicitaire CDN n'est pas configuré : les segments publicitaires utilisent l'emplacement direct de MediaTailor comme URL de base.
- Préfixe de segment de contenu CDN configuré : les segments de contenu utilisent le préfixe CDN comme URL de base.
- Le préfixe du segment de contenu CDN n'est pas configuré : les segments de contenu font référence à l'origine du contenu d'origine.

Gestion des URL DASH BaseURL

Pour les manifestes DASH, MediaTailor gère BaseURL les paramètres différemment pour le contenu et les segments publicitaires :

Segments de contenu :

- Avec le préfixe de segment de contenu CDN : MediaTailor définit exactement un BaseURL au MPD niveau en utilisant le préfixe que vous avez spécifié.
- Sans préfixe de segment de contenu CDN : MediaTailor préserve BaseURL les paramètres existants du manifeste d'origine ou en ajoute un en fonction de l'MPDURL d'origine s'il n'en existe aucun.

Segments publicitaires :

- Avec le préfixe de segment publicitaire CDN : chaque période d'annonce obtient exactement une BaseURL en utilisant le préfixe configuré.
- Sans préfixe de segment publicitaire CDN : chaque période publicitaire reçoit exactement un préfixe BaseURL pointant vers le serveur de contenu MediaTailor publicitaire.

Exemple Exemple de mappage CDN

Si l'origine de votre contenu est `http://origin.com/contentpath/` et le préfixe de segment de contenu de votre CDN l'est `https://cdn.example.com/`, alors un segment de contenu qui devrait normalement être référencé tel qu'il `http://origin.com/contentpath/subdir/content.ts` apparaîtra dans le manifeste sous la forme `https://cdn.example.com/subdir/content.ts`

Considérations importantes

Lorsque vous configurez le mappage du CDN dans MediaTailor, gardez à l'esprit les considérations importantes suivantes :

Utiliser le protocole HTTPS pour les préfixes du CDN

Utilisez toujours le protocole HTTPS URLs pour vos préfixes de CDN afin de garantir une diffusion de contenu sécurisée.

Faites correspondre les chemins de comportement du CDN

Assurez-vous que les préfixes de CDN que vous configurez MediaTailor correspondent aux modèles de chemin que vous avez configurés dans vos comportements de CDN.

Tenez compte du régional CDNs

Si vous utilisez différents domaines CDN pour différentes régions, vous devrez créer des MediaTailor configurations distinctes pour chaque région.

Vérifier la propriété du domaine

Assurez-vous de contrôler les domaines CDN dans lesquels vous configurez. MediaTailor

Vérifier la configuration du mappage du CDN

Après avoir configuré le mappage du CDN dans MediaTailor, vérifiez que votre configuration fonctionne correctement :

1. Demandez un manifeste via votre CDN.
2. Examinez le contenu du manifeste pour vérifier que le segment fait URLs référence à votre domaine CDN.
3. Vérifiez que le segment URLs de contenu du manifeste pointe vers votre domaine CDN.
4. Vérifiez que le segment URLs publicitaire du manifeste pointe vers votre domaine CDN.

Pour des procédures complètes de test et de validation, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#).

Étapes suivantes

Après avoir configuré le mappage du CDN MediaTailor, l'étape suivante consiste à mettre en œuvre les meilleures pratiques de sécurité pour votre intégration du CDN. Pour obtenir des instructions, consultez [Bonnes pratiques de sécurité de l'intégration CDN pour MediaTailor](#).

Bonnes pratiques de sécurité de l'intégration CDN pour MediaTailor

Cette section explique comment mettre en œuvre les meilleures pratiques de sécurité pour l'intégration de votre réseau de diffusion de AWS Elemental MediaTailor contenu (CDN). Une configuration de sécurité appropriée permet de protéger votre contenu et votre infrastructure contre les accès non autorisés et les menaces potentielles.

La sécurisation de votre intégration CDN est essentielle pour protéger votre contenu, empêcher les accès non autorisés et garantir le respect des exigences de sécurité. La mise en œuvre de ces meilleures pratiques permet de créer une posture de sécurité robuste pour votre flux de travail de streaming.

Configuration de sécurité du CDN

Avant de configurer les paramètres de routage et de cache de votre CDN, mettez en œuvre les meilleures pratiques de sécurité suivantes pour protéger votre contenu et votre infrastructure :

Sûreté des transports

MediaTailor utilise uniquement le protocole HTTPS pour toutes les communications et n'autorise pas les connexions HTTP. Pour garantir une communication sécurisée entre tous les composants de votre flux de travail de streaming, procédez comme suit :

1. Configurez le protocole HTTPS pour toutes les communications entre votre CDN et les points de MediaTailor terminaison. MediaTailor nécessite le protocole HTTPS et n'accepte pas les connexions HTTP.
2. Utilisez le protocole TLS 1.2 ou version ultérieure pour toutes les connexions HTTPS.
3. Configurez votre CDN pour appliquer les connexions HTTPS uniquement aux spectateurs.

Contrôle d'accès

Pour protéger votre contenu et votre infrastructure contre tout accès non autorisé, mettez en œuvre les contrôles d'accès suivants :

1. Configurez les paramètres de géo-restriction si vous devez limiter l'accès au contenu à des régions spécifiques.
2. Implémentez des cookies signés URLs ou signés pour le contenu qui nécessite l'authentification du spectateur.
3. Configurez des listes d'autorisation basées sur l'adresse IP pour l'accès administratif à la configuration du CDN.

Surveillance de la sécurité

Pour détecter les événements de sécurité et y répondre efficacement, mettez en œuvre les pratiques de surveillance suivantes :

1. Activez la journalisation des accès pour votre distribution CDN.
2. Configurez des alertes en cas de modèles de trafic inhabituels ou de tentatives d'accès.
3. Passez régulièrement en revue les configurations de sécurité et mettez-les à jour si nécessaire.

Fonctionnalités de sécurité avancées

Les fonctionnalités de sécurité avancées suivantes fournissent une protection améliorée pour votre intégration au CDN :

Pare-feu d'application Web (WAF)

Si votre CDN prend en charge la fonctionnalité WAF, configurez-le pour le protéger contre les vulnérabilités et attaques Web courantes.

DDoProtection S

Activez DDo les fonctionnalités de protection S fournies par votre CDN pour atténuer les denial-of-service attaques distribuées.

Contrôle d'accès à l'origine

Configurez vos serveurs d'origine pour accepter uniquement les demandes provenant de votre CDN afin d'empêcher les tentatives d'accès direct.

Chiffrement du contenu

Pour les contenus très sensibles, envisagez de mettre en œuvre des mécanismes de chiffrement supplémentaires au-delà de la sécurité du transport.

Étapes suivantes

Après avoir mis en œuvre les meilleures pratiques de sécurité, l'étape suivante consiste à tester et à résoudre les problèmes liés à votre intégration au CDN. Consultez [Résoudre les problèmes liés à l'intégration du CDN](#) les instructions complètes de test et de dépannage.

Configurez SSAI avec un CDN pour une publicité vidéo personnalisée

Cette section fournit des conseils complets sur l'intégration de l'insertion publicitaire AWS Elemental MediaTailor côté serveur (SSAI) à un réseau de diffusion de contenu (CDN). Suivez ces étapes pour configurer, configurer et optimiser votre intégration SSAI CDN.

L'insertion de publicités côté serveur (SSAI) est une technologie qui insère de manière fluide des publicités personnalisées dans les flux vidéo au niveau du serveur plutôt qu'au niveau du client. Associé à un CDN, vous créez une solution robuste et évolutive pour diffuser des publicités personnalisées à un public mondial avec une latence minimale.

Dans cette rubrique, nous utilisons le terme manifestes pour désigner collectivement les listes de lecture multivariantes, les listes de lecture multimédia et MPDs.

Ce dont vous aurez besoin

Avant de configurer l'insertion d' MediaTailor annonces avec un CDN, rassemblez les ressources requises suivantes :

AWS compte et autorisations

Un AWS compte doté des autorisations appropriées pour créer et gérer MediaTailor des ressources

Autorisations IAM pour MediaTailor, CloudFront (si vous en utilisez), et les services connexes

Pour connaître les exigences détaillées en matière d'autorisation, voir [Sécurité dans AWS Elemental MediaTailor](#).

Services requis

AWS Elemental MediaTailor configuré et en cours d'exécution

Un compte de réseau de diffusion de contenu (CDN) (Amazon CloudFront ou CDN tiers)

Serveur d'origine pour votre contenu (HLS ou DASH)

Serveur de décision publicitaire (ADS) compatible avec VAST ou VMAP

Exigences relatives au contenu

Contenu correctement encodé et emballé au format HLS ou DASH

Des marqueurs de coupure de publicité dans votre contenu (pour la VOD) ou des marqueurs SCTE-35 (pour le direct)

Avant de commencer

Effectuez les tâches de configuration suivantes avant de mettre en œuvre l'insertion de MediaTailor publicités avec un CDN :

1. Configurer la connectivité réseau entre votre CDN et MediaTailor les serveurs d'origine
2. Configurer le protocole HTTPS pour une diffusion de contenu sécurisée
3. Configurer les paramètres DNS pour votre domaine CDN
4. Pour la MediaTailor configuration de base, suivez les étapes décrites dans [Configuration et Commencer à insérer MediaTailor des publicités](#).

Connaissances requises

Pour implémenter cette solution avec succès, vous devez disposer des éléments suivants :

- Compréhension des protocoles de streaming (HLS/DASH)
- Connaissances de base des principes de configuration du CDN
- Connaissance des concepts d'insertion de publicités

Avantages de l'intégration du CDN

L'intégration de SSAI à un CDN offre les principaux avantages suivants :

Expérience utilisateur améliorée

Une configuration correcte CDNs réduit la mise en mémoire tampon, les temps de démarrage et les erreurs de lecture lors des transitions publicitaires. Cela se traduit par un engagement et une satisfaction accrus des spectateurs.

Réduction des coûts

Des stratégies de mise en cache efficaces minimisent les demandes d'origine. Cela réduit les coûts de transfert de données et la charge du serveur d'origine, ce qui est particulièrement important pour les gros volumes de contenu financé par la publicité.

Capacité de mise à l'échelle

Les configurations CDN optimisées permettent de gérer les pics de trafic lors d'événements populaires sans dégrader les performances, garantissant ainsi la diffusion de vos publicités personnalisées même aux heures de pointe.

Portée mondiale

Configuré correctement, CDNs diffusez du contenu avec une faible latence aux spectateurs du monde entier, où qu'ils se trouvent, élargissant ainsi votre audience potentielle.

Transitions publicitaires fluides

La configuration optimisée du CDN garantit des transitions fluides entre le contenu et les publicités, créant ainsi une expérience de visionnage de qualité télévisuelle.

Les rubriques suivantes fournissent des conseils complets sur la configuration à l' MediaTailor aide d'un CDN pour des performances optimales.

Rubriques

- [Comprendre l'architecture d'insertion publicitaire pour le CDN et les intégrations MediaTailor](#)
- [Configurez un MediaTailor SSAI de base avec un CDN pour une diffusion publicitaire optimale](#)
- [Intégrez le MediaTailor SSAI à l'assemblage des canaux pour les canaux linéaires monétisés](#)
- [Optimisez les performances du CDN et MediaTailor des intégrations](#)
- [Surveillez les opérations pour le CDN et MediaTailor les intégrations](#)
- [Résoudre les problèmes liés à MediaTailor SSAI CDNs pour une diffusion ininterrompue des publicités](#)

Comprendre l'architecture d'insertion publicitaire pour le CDN et les intégrations MediaTailor

Cette section explique les concepts et l'architecture de l'insertion de publicités côté serveur (SSAI) avec des réseaux de diffusion de contenu (CDNs) pour. AWS Elemental MediaTailor Vous découvrirez comment l'insertion dynamique d'annonces et la manipulation de manifestes fonctionnent ensemble pour permettre une monétisation vidéo efficace.

L'insertion de publicités côté serveur (SSAI) MediaTailor vous permet de :

- Insérez de la publicité personnalisée dans vos flux vidéo à des points de rupture publicitaires définis
- Cibler les publicités avec précision en fonction des données des spectateurs
- Éliminez le besoin d'une technologie d'insertion publicitaire côté client

Associés à un CDN, vous pouvez diffuser ces flux personnalisés aux spectateurs avec des performances et une évolutivité améliorées, améliorant ainsi votre stratégie de monétisation vidéo.

L'architecture recommandée pour l'insertion d'annonces avec un CDN positionne le CDN entre les spectateurs et l'insertion d'annonces, l'insertion d'annonces accédant au contenu directement depuis votre origine. Cette architecture offre les avantages suivants en termes de diffusion de contenu et de monétisation vidéo :

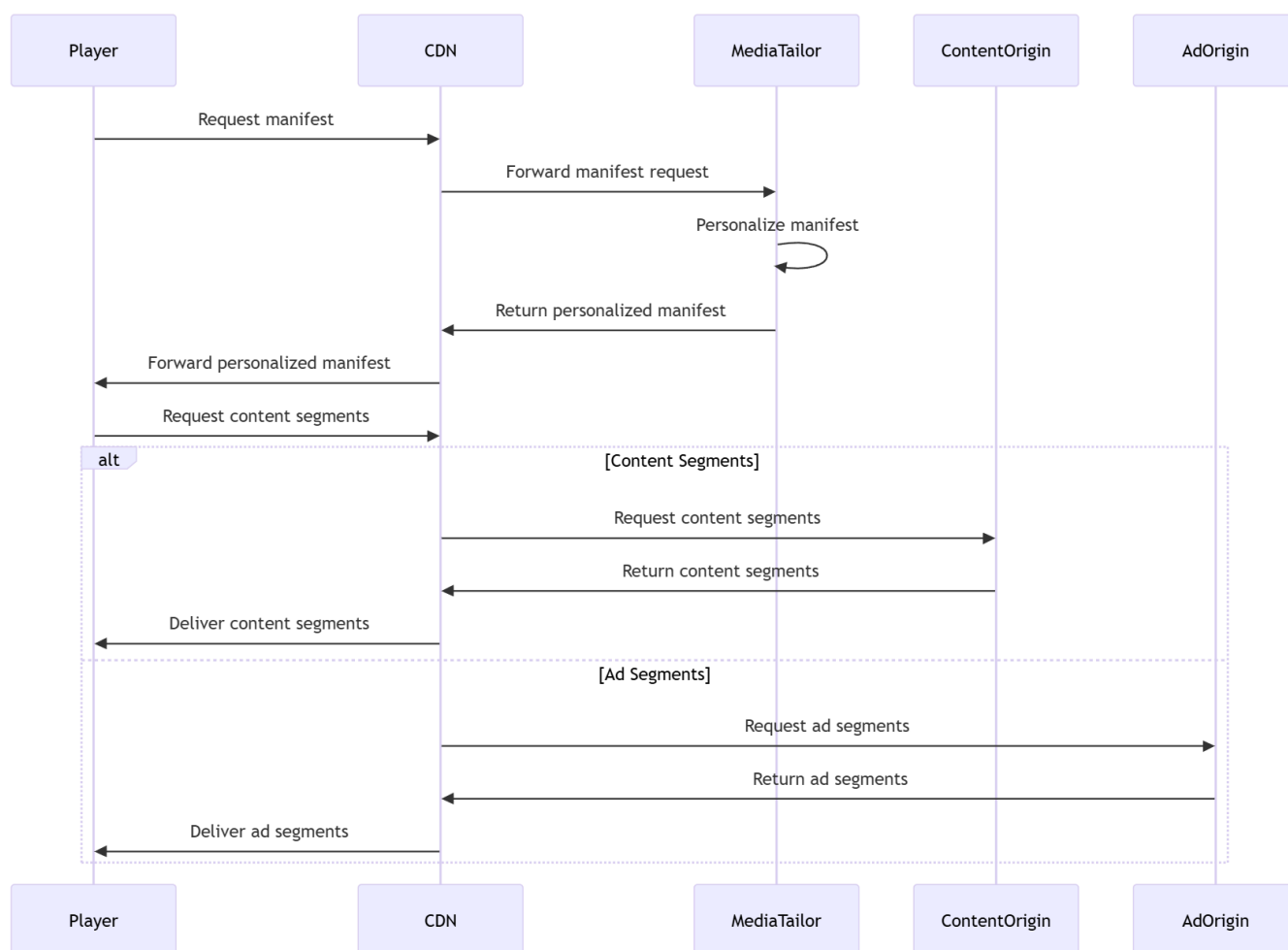
- Mise en cache efficace du contenu et des segments publicitaires
- Charge de demande réduite sur MediaTailor
- Vitesse de diffusion améliorée pour les spectateurs

- Gestion simplifiée des URL
- Diffusion cohérente de publicités personnalisées sur tous les appareils

Dans cette architecture recommandée :

1. Les spectateurs demandent des manifestes au CDN
2. Le CDN transmet les demandes à l'insertion d'annonces
3. Les demandes d'insertion d'annonces (contenu, manifestes) dès l'origine
4. L'insertion d'annonces demande des publicités au serveur de décision publicitaire (ADS)
5. L'insertion d'annonces personnalise les manifestes en remplaçant les marqueurs publicitaires (du manifeste d'origine) par ceux URLs qui pointent vers des segments publicitaires ciblés pour un internaute spécifique (issus de l'ADS)
6. L'insertion d'annonces renvoie les manifestes personnalisés contenant le segment publicitaire URLs au CDN, qui les transmet aux spectateurs
7. Les spectateurs demandent des segments via le CDN
8. Le CDN achemine les demandes de segment en fonction du type de segment :
 - Les demandes de segment de contenu sont dirigées vers l'origine du contenu
 - Les demandes de segments publicitaires sont envoyées à MediaTailor

Cette architecture garantit des performances optimales tout en conservant les avantages de sécurité et de flexibilité liés à l'utilisation d'un CDN.



Note

Ce flux varie légèrement entre la VOD et le contenu en direct. Pour la VOD, les manifestes peuvent être mis en cache plus longtemps, tandis que le contenu en direct nécessite des mises à jour plus fréquentes des manifestes pour assurer la continuité des flux.

La principale différence entre la VOD et la mise en cache de contenu en direct :

Contenu VOD

Définissez des valeurs TTL plus longues (minutes à heures) pour les manifestes, car elles ne changent pas fréquemment

Contenu en direct

Définissez des valeurs TTL plus courtes (secondes) pour les manifestes afin de garantir que les spectateurs reçoivent les segments de diffusion les plus récents

Nous vous déconseillons de placer un CDN entre l'origine de votre contenu et AWS Elemental MediaTailor. Cela peut entraîner plusieurs défis techniques :

Collisions entre clés de cache

Configurez votre CDN pour qu'il gère correctement les paramètres de requête. Cela permet MediaTailor d'éviter de recevoir des manifestes incorrects lorsque vous demandez le même manifeste avec des paramètres de requête différents.

Problèmes de compression Gzip

Si vous rencontrez des erreurs d'analyse de manifestes, assurez-vous que votre CDN fournit des manifestes correctement formatés à. MediaTailor Certains CDNs peuvent fournir des charges utiles gzip corrompues susceptibles de provoquer des échecs d'analyse. Dans ce cas, vous devrez peut-être désactiver la compression entre votre CDN et MediaTailor tout en maintenant la compression afin de réaliser des économies à d'autres étapes de votre flux de travail.

Fraîcheur manifeste

Pour les diffusions en direct, configurez votre CDN pour qu'il envoie les manifestes actuels à MediaTailor. Cela permet d'éviter les problèmes de synchronisation entre le contenu et les publicités.

Optimisation des performances

Minimisez les sauts sur le réseau et les pertes de cache potentielles afin de réduire les temps de démarrage de la lecture.

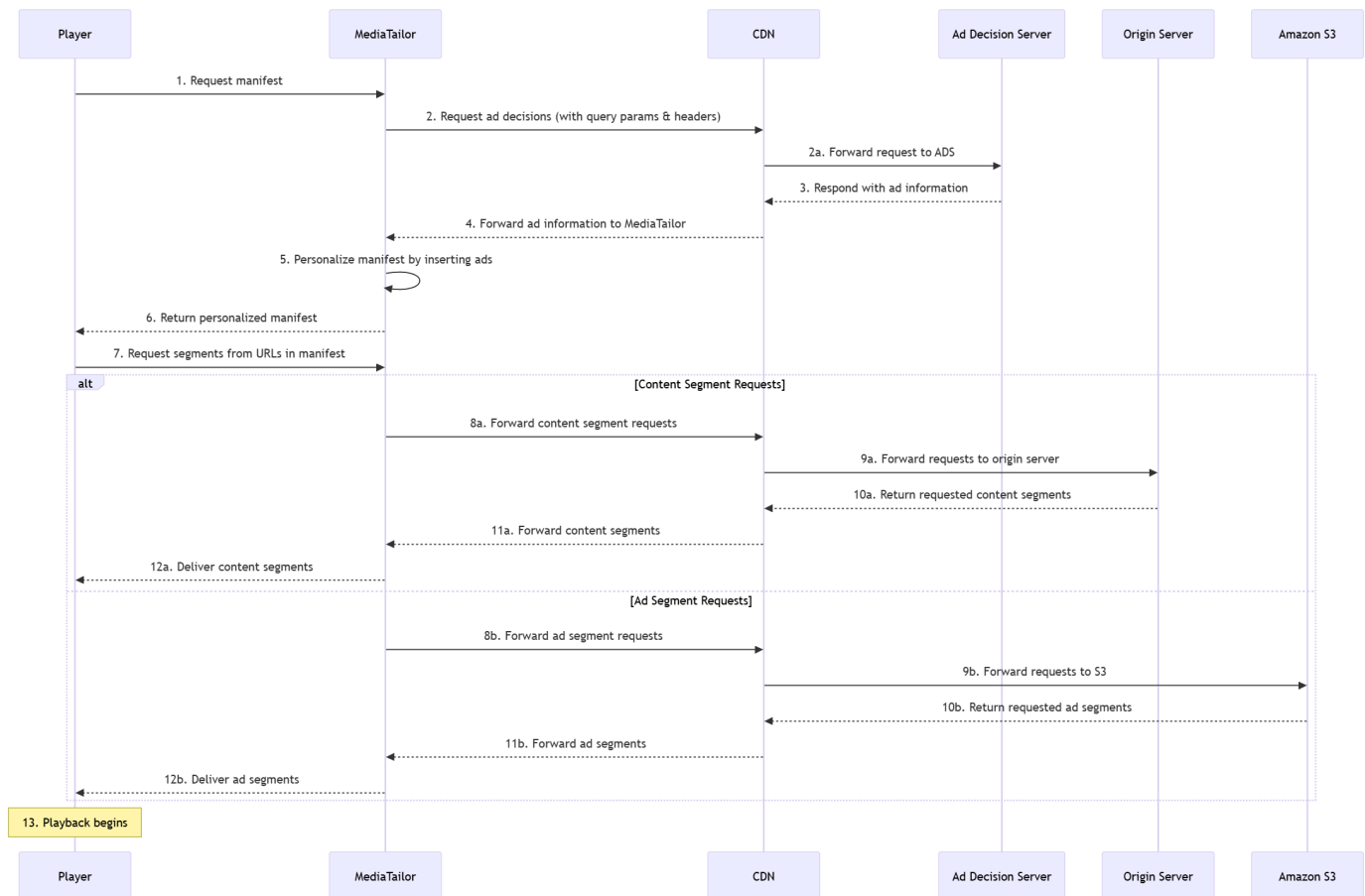
Gestion du cache

Mettez en œuvre des stratégies simplifiées d'invalidation du cache, en particulier pour le contenu en direct où les manifestes sont fréquemment mis à jour.

Dans cette architecture sous-optimale :

1. Les spectateurs demandent des playlists multivariantes, des playlists multimédias ou MPDs directement auprès de AWS Elemental MediaTailor.

2. MediaTailor demande des manifestes de contenu (listes de lecture multivariantes, listes de lecture multimédia ou MPDs) via le CDN.
3. Le CDN transmet les demandes au serveur d'origine.
4. Le serveur d'origine renvoie des listes de lecture multivariantes, des listes de lecture multimédia ou MPDs vers le CDN.
5. Le CDN transmet des listes de lecture multivariantes, des listes de lecture multimédia ou MPDs vers MediaTailor.
6. MediaTailor demande des publicités au serveur de décision publicitaire (ADS).
7. MediaTailor personnalise les manifestes en insérant des publicités dans des playlists multivariantes, des playlists multimédias ou MPDs en les diffusant directement aux spectateurs.
8. Cette architecture introduit une latence supplémentaire, des problèmes potentiels de mise en cache et complique le dépannage.



Flux de demandes et de réponses

Lorsque vous implémentez l'insertion dynamique d'annonces avec un CDN, configurez votre système pour prendre en charge ce flux de demandes et de réponses :

1. Configurez votre lecteur pour qu'il demande des playlists multivariantes (HLS) ou MPDs (DASH) depuis votre CDN avec MediaTailor comme origine manifeste.
2. Configurez votre CDN pour qu'il transfère toutes les demandes de playlist multivariante, de playlist multimédia et de MPD MediaTailor, y compris tous les paramètres de requête et les en-têtes.
3. Assurez-vous de MediaTailor pouvoir communiquer avec votre serveur de décision publicitaire (ADS) en lui transmettant les paramètres de requête et les en-têtes.
4. Configurez votre ADS pour utiliser les paramètres de requête afin de déterminer les annonces à insérer.

5. Configurez le préfixe CDN dans la configuration de MediaTailor lecture afin de MediaTailor pouvoir remplacer les noms de domaine du CDN par des préfixes d'URL de contenu et de segment publicitaire.
6. Configurez votre CDN pour transférer des listes de lecture multivariantes personnalisées, des listes de lecture multimédia et MPDs depuis MediaTailor le lecteur demandeur.
7. Configurez votre CDN pour traduire le segment URLs, en transférant les demandes de segment de contenu vers le serveur d'origine et les demandes publicitaires vers le compartiment Amazon S3 où sont stockées MediaTailor les publicités transcodées.

Terminologie CDN pour l'insertion de publicités

La compréhension de ces termes clés vous aidera à implémenter et à résoudre les problèmes liés à l'intégration de votre CDN d'insertion publicitaire :

CDN d'origine et CDN Edge

CDN d'origine : un CDN positionné entre MediaTailor et l'origine de votre contenu. Il met en cache les segments de contenu afin de réduire la charge sur vos serveurs d'origine. Dans une architecture multi-CDN, il s'agit de la première couche CDN qui s'interface directement avec l'origine.

Edge CDN : un CDN positionné entre les spectateurs et MediaTailor. Il fournit des manifestes et du contenu personnalisés aux spectateurs. Dans une architecture multi-CDN, il s'agit de la couche CDN la plus externe qui s'interface directement avec les spectateurs.

Termes de configuration du CDN

Comportement du cache : règles qui déterminent la manière dont un CDN gère les différents types de demandes. Ces règles incluent :

- Paramètres de durée de mise en cache
- Configurations de routage d'origine
- Paramètres de traitement des demandes

TTL (Time To Live) : durée pendant laquelle le contenu reste valide dans un cache CDN avant de devoir être actualisé depuis son origine.

Clé de cache : identifiant unique utilisé par un CDN pour stocker et récupérer le contenu mis en cache. Il inclut généralement :

- chemin de l'URL
- Paramètres Query (Requête)
- En-têtes sélectionnés

Origin Shield : couche de mise en cache intermédiaire entre les emplacements périphériques du CDN et votre serveur d'origine. Cela réduit le nombre de demandes adressées à votre origine.

Réduction des demandes : fonctionnalité CDN qui combine plusieurs demandes simultanées pour le même contenu en une seule demande d'origine.

MediaTailor-termes spécifiques au CDN

Préfixe de segment de contenu CDN : nom de domaine CDN AWS Elemental MediaTailor utilisé lors de la génération de segments de contenu dans URLs des manifestes.

Préfixe de segment publicitaire CDN : nom de domaine CDN MediaTailor utilisé lors de la génération de segments publicitaires dans URLs les manifestes.

Pour plus d'informations sur la configuration du CDN avec MediaTailor, consultez [Configurer l'intégration du CDN](#).

 Note

Ces termes sont cohérents avec ceux utilisés dans la documentation d'assemblage des canaux. Pour la terminologie de l'assemblage des canaux, voir [Terminologie CDN pour l'assemblage de canaux](#).

Configurez un MediaTailor SSAI de base avec un CDN pour une diffusion publicitaire optimale

Cette section fournit des step-by-step instructions pour configurer l'insertion AWS Elemental MediaTailor dynamique de publicités avec un réseau de diffusion de contenu (CDN) afin d'optimiser votre flux de monétisation vidéo.

Pour une configuration avancée du serveur publicitaire à l'aide de variables dynamiques, voir [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour plus d'informations sur le transfert de paramètres CDNs pour le ciblage publicitaire, consultez [MediaTailor paramètres de requête manifeste](#).

Pour des informations conceptuelles sur le SSAI avec CDN, voir [Comprendre l'architecture d'insertion publicitaire pour le CDN et les intégrations MediaTailor](#).

Conditions préalables

Avant de configurer l'insertion d'annonces avec un CDN, assurez-vous de disposer des éléments suivants :

- Une AWS Elemental MediaTailor configuration active
- Un serveur d'origine de contenu diffusant du contenu HLS ou DASH avec des marqueurs publicitaires appropriés

Pour plus d'informations sur les marqueurs publicitaires, consultez [Comprendre le comportement d'insertion des publicités](#).

- Un serveur de décision publicitaire (ADS) compatible avec VAST ou VMAP pour le ciblage publicitaire
- Un compte CDN (tel qu'Amazon CloudFront ou un autre fournisseur de CDN)
- Connaissances de base de la manipulation des manifestes et des concepts d'insertion dynamique de publicités

Étape 1 : configurer la mise en cache du CDN pour une diffusion publicitaire optimale

Une configuration correcte de la mise en cache du CDN est essentielle pour optimiser les performances de votre flux de travail de monétisation vidéo. Les exigences de mise en cache diffèrent entre l'insertion publicitaire côté serveur (SSAI) et l'insertion publicitaire guidée par le serveur (SGAI). Utilisez ces paramètres recommandés pour garantir une diffusion efficace du contenu et de la publicité personnalisée :

Paramètres de mise en cache SSAI CDN

Pour les flux de travail d'insertion de publicités côté serveur, une configuration de mise en cache appropriée est essentielle pour des performances optimales. Le SSAI nécessite des valeurs TTL spécifiques et des paramètres de clé de cache pour garantir que les manifestes personnalisés ne sont pas mis en cache alors que les segments sont mis en cache efficacement.

Pour connaître les paramètres de mise en cache SSAI détaillés, notamment les valeurs TTL, les modèles de chemin et les configurations des clés de cache, consultez le guide d'[Mise en cache d'insertion de publicités côté serveur \(SSAI\)](#) optimisation du CDN.

Principaux principes de mise en cache pour le SSAI :

- Manifestes : définissez le TTL sur 0 seconde pour empêcher la mise en cache du contenu personnalisé
- Segments : mise en cache agressive (plus de 24 heures) pour réduire la charge d'origine
- Clés de cache : inclure tous les paramètres de requête pour les manifestes, le chemin d'URL uniquement pour les segments

Paramètres de mise en cache du CDN SGAI

Pour les flux de travail d'insertion de publicités guidés par le serveur, les exigences de mise en cache diffèrent de celles du SSAI, car les manifestes SGAI peuvent être mis en cache pendant de courtes périodes tout en proposant des expériences publicitaires personnalisées.

Pour connaître les paramètres complets de mise en cache SGAI, y compris les valeurs VOD et TTL en direct, consultez les tableaux de mise en cache du guide d'optimisation. Le SGAI permet une meilleure efficacité du cache que le SSAI tout en conservant les fonctionnalités de personnalisation des publicités.

Principales différences de mise en cache SGAI :

- Manifestes : peuvent être mis en cache pendant de courtes périodes (5 à 30 minutes pour la VOD, 2 à 10 secondes pour le live)
- Segments : mettez en cache de manière agressive, comme SSAI (plus de 24 heures pour la plupart des contenus)
- Avantage en termes de performances : meilleurs ratios d'accès au cache que le SSAI grâce aux manifestes pouvant être mis en cache

Pour Amazon CloudFront, vous pouvez implémenter ces paramètres en utilisant des comportements de cache avec différentes valeurs TTL et politiques relatives aux clés de cache. Pour les autres CDNs, reportez-vous à leur documentation spécifique pour implémenter des règles de mise en cache similaires.

Étape 2 : Mettre en œuvre des approches hybrides (si nécessaire)

Si votre architecture nécessite une approche hybride avec un CDN ou une couche de mise en cache distinct entre l'origine du contenu et : MediaTailor

1. Mettez en place une séparation claire des préoccupations dans la configuration de votre CDN.
2. Configurez des paramètres CDN spécifiques pour éviter les problèmes techniques décrits dans la section précédente.
3. Testez soigneusement votre configuration pour vérifier que la personnalisation du manifeste fonctionne correctement.
4. Surveillez les indicateurs de performance pour garantir une diffusion optimale des playlists multivariantes, des playlists multimédias et des segments. MPDs

Lors de la mise en œuvre d'une approche hybride, tenez compte des configurations spécifiques suivantes :

- Pour le CDN entre l'origine du contenu et MediaTailor :
 - Configurer le transfert de compression pour les fichiers manifestes afin de préserver l'état de compression d'origine
 - Inclure tous les paramètres de requête dans la clé de cache
 - Définissez des valeurs TTL courtes pour les manifestes de contenu en direct
- Pour le CDN entre utilisateurs MediaTailor et spectateurs :
 - Configurez des temps de cache plus longs pour les segments publicitaires
 - Set approprié TTLs pour les manifestes personnalisés
 - Mettez en œuvre un routage d'origine approprié pour le contenu par rapport aux segments publicitaires

Étape 3 : terminer la configuration de votre CDN

Après avoir choisi votre architecture et compris le flux de demandes, terminez votre configuration en suivant les étapes de configuration détaillées décrites dans [Configurer l'intégration du CDN](#).

Pour les fournisseurs de CDN spécifiques, consultez ces ressources supplémentaires :

- Amazon CloudFront : voir [intégration d'CloudFront](#) pour les CloudFront étapes de configuration spécifiques
- Autre CDNs : appliquez les principes généraux décrits dans ce guide, en les adaptant aux options de configuration spécifiques de votre CDN

Étape 4 : Vérifiez votre configuration

Après avoir terminé la configuration de votre CDN, vérifiez que votre flux de travail d'insertion d'annonces dynamique fonctionne correctement :

1. Testez la lecture via votre CDN avec un lecteur d'échantillons
2. Vérifiez que la publicité personnalisée est correctement insérée aux points de rupture publicitaires désignés
3. Vérifiez les journaux du CDN pour confirmer le bon routage des demandes
4. Surveillez les taux de réussite du cache pour garantir des performances optimales à la fois pour le contenu et les segments publicitaires
5. Vérifiez que les paramètres de ciblage publicitaire sont correctement transmis dans le flux de travail

Pour des procédures complètes de test et de validation, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#). Pour des informations détaillées sur le suivi de la mise en œuvre de votre SSAI, consultez [Surveillez les opérations pour le CDN et MediaTailor les intégrations](#). Pour optimiser les performances, voir [Optimisez les performances du CDN et MediaTailor des intégrations](#).

Intégrez le MediaTailor SSAI à l'assemblage des canaux pour les canaux linéaires monétisés

Cette rubrique explique comment associer l'insertion publicitaire AWS Elemental MediaTailor côté serveur à l'assemblage de canaux et à l'intégration d'un réseau de diffusion de contenu (CDN). Cette intégration vous permet de :

- Créez des canaux linéaires monétisés avec de la publicité personnalisée
- Diffusez des publicités ciblées auprès de différents spectateurs qui regardent le même contenu
- Préservez des expériences de visionnage de qualité audiovisuelle

Avantages de la combinaison du SSAI avec l'assemblage de canaux

L'intégration du SSAI à l'assemblage des canaux offre plusieurs avantages clés :

Monétisation des canaux linéaires

Insérez des publicités personnalisées dans vos canaux linéaires pour générer des revenus à partir de votre bibliothèque de contenu. Vous pouvez monétiser le contenu en direct et le contenu VOD au sein d'un seul flux linéaire.

Publicité personnalisée

Diffusez différentes publicités à différents spectateurs qui regardent le même contenu de chaîne. Cette approche ciblée augmente la pertinence des publicités et les revenus potentiels par rapport à la publicité diffusée traditionnelle.

Gestion simplifiée des interruptions publicitaires

Définissez les points de rupture des publicités dans vos programmes d'assemblage de chaînes sans avoir à conditionner le contenu à l'aide de marqueurs SCTE-35. Il est ainsi plus facile d'insérer des publicités aux points de rupture naturels de votre contenu.

Une expérience de qualité audiovisuelle

MediaTailor garantit une expérience de visionnage de haute qualité grâce à des transitions fluides entre le contenu et les publicités. L'insertion de publicités côté serveur élimine de nombreux problèmes courants tels que :

- Mise en mémoire tampon pendant les transitions publicitaires
- Des bloqueurs de publicités empêchant la monétisation
- Qualité de lecture incohérente

Livraison évolutive

Associée à un CDN, cette intégration peut être étendue à des millions de spectateurs simultanés sans dégradation des performances ou des capacités de personnalisation.

Présentation de l'architecture

L'architecture permettant de combiner le SSAI avec l'assemblage de canaux implique généralement les composants suivants :

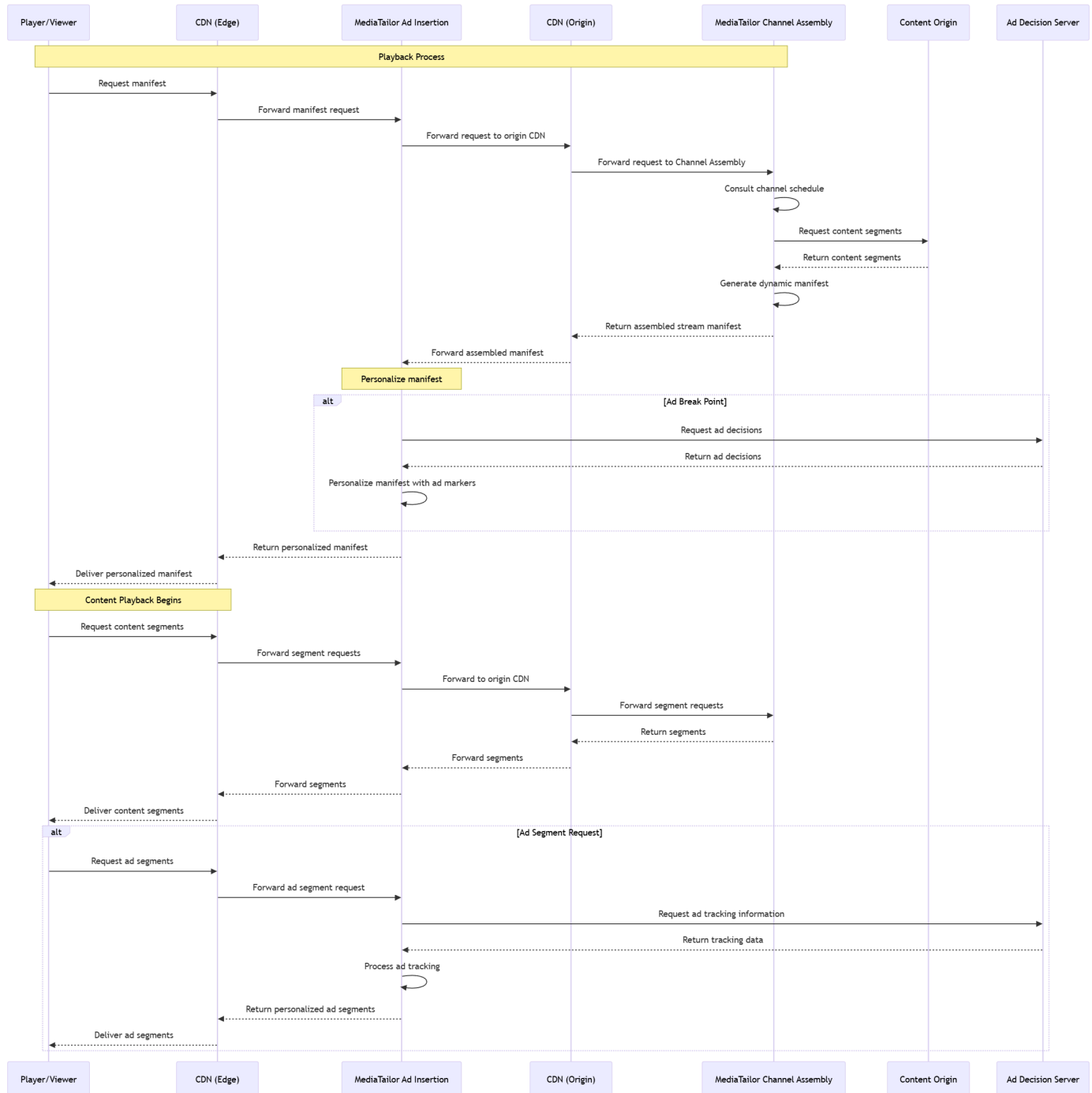
- Assemblage de chaînes : crée des chaînes linéaires à partir de VOD et de contenu en direct, et insère du contenu en ardoise qui crée des marqueurs publicitaires dans le manifeste généré
- Insertion d'annonces : reconnaît les points de rupture des annonces et insère des éléments URLs pointant vers des segments publicitaires personnalisés dans le manifeste

- Serveur de décision publicitaire (ADS) : détermine les publicités à insérer pour chaque spectateur
- Réseau de diffusion de contenu (CDN) : fournit le contenu assemblé et les segments publicitaires aux spectateurs
- Serveur d'origine : stocke les segments de VOD et de contenu en direct

Dans cette architecture :

1. L'assemblage des chaînes crée une chaîne linéaire à partir de votre contenu VOD et en direct, et insère du contenu en ardoise qui crée les marqueurs publicitaires dans le manifeste généré
2. Lorsqu'un internaute demande la chaîne, l'insertion d'annonces reconnaît les pauses publicitaires insérées dans la chaîne linéaire
3. L'insertion d'annonces appelle l'ADS pour recevoir la liste des publicités, les transcode et les insère URLs pointant vers les segments d'annonces transcodés dans le manifeste personnalisé
4. Le CDN fournit le flux personnalisé au spectateur

Le schéma suivant illustre ce flux de travail :



Configuration de l'intégration

Pour configurer le SSAI avec assemblage de canaux :

1. Configurez votre CDN Edge pour accepter les demandes de manifeste des spectateurs et les transférer vers l'insertion AWS Elemental MediaTailor publicitaire.
2. Configurez l'insertion d' MediaTailor annonces pour transférer les demandes vers votre CDN d'origine.
3. Configurez votre CDN d'origine pour transmettre les demandes à l'assemblage des MediaTailor canaux.
4. Configurez l'assemblage des MediaTailor canaux pour générer des manifestes dynamiques en fonction du calendrier actuel.
5. Configurez votre CDN d'origine pour transférer les manifestes assemblés vers l'insertion MediaTailor publicitaire.
6. Configurez l'insertion d' MediaTailor annonces pour demander des décisions publicitaires à votre serveur de décision publicitaire aux points de rupture des publicités.
7. Configurez l'insertion d' MediaTailor annonces pour personnaliser les manifestes en remplaçant les marqueurs publicitaires (issus de l'assemblage des chaînes) par des URLs signaux pointant vers des segments publicitaires ciblés (issus de l'ADS).
8. Configurez votre CDN Edge pour fournir des manifestes personnalisés aux spectateurs.
9. Configurez votre architecture CDN pour gérer efficacement les demandes de contenu et de segments publicitaires.

Définition des pauses publicitaires dans l'assemblage des chaînes

Lorsque vous créez des programmes dans le cadre de l'assemblage de chaînes, vous pouvez définir les pauses publicitaires de plusieurs manières :

Transitions entre programmes

Insérez des publicités entre les programmes dans le calendrier de vos chaînes. Il s'agit de l'approche la plus simple qui garantit que les publicités n'interrompent pas le contenu du programme.

Marqueurs SCTE-35

Si votre contenu VOD contient des marqueurs SCTE-35, l'assemblage des chaînes peut préserver ces marqueurs, et l'insertion de publicités peut les utiliser comme points de rupture publicitaire.

Insertion basée sur le temps

Définissez des pauses publicitaires à des moments précis au sein des programmes. Cela vous permet d'insérer des publicités à des points de rupture naturels dans votre contenu.

Pour obtenir des informations détaillées sur la création de programmes comportant des interruptions publicitaires, consultez la section [Utilisation des programmes](#).

Considérations relatives à la mise en cache du CDN

Pour des performances optimales lorsque vous combinez l'assemblage de canaux et le SSAI avec un CDN :

- Configurer des comportements de cache qui font la distinction entre l'assemblage de canaux et les demandes SSAI
- Définissez les valeurs TTL appropriées pour les manifestes et les segments, comme recommandé dans [Étape 1 : configurer la mise en cache du CDN pour une diffusion publicitaire optimale](#)
- Garantisiez un routage correct entre l'assemblage des chaînes, l'insertion de publicités et les origines de votre CDN
- Surveillez les indicateurs de performance pour les composants d'assemblage des chaînes et d'insertion de publicités

Paramètres de mise en cache recommandés pour une mise en œuvre combinée

Type de contenu	TTL	Éléments clés du cache
Manifestes d'assemblage des canaux	0 secondes	Chemin d'URL et paramètres de requête
Manifestes personnalisés SSAI	0 secondes	Chemin d'URL + tous les paramètres de requête
Segments de contenu	24 heures et plus	Chemin d'URL uniquement
Segments publicitaires	24 heures et plus	Chemin d'URL uniquement

Surveillance de la solution intégrée

Pour vous assurer que votre solution intégrée fonctionne de manière optimale, surveillez ces indicateurs clés :

Métriques d'assemblage des canaux

Surveillez le temps de génération du manifeste, les transitions entre les programmes et les éventuelles erreurs dans le processus d'assemblage des canaux.

Mesures d'insertion de publicités

Suivez le taux de remplissage des annonces, les temps de réponse du serveur de décision publicitaire et les erreurs d'insertion d'annonces.

Métriques du CDN

Surveillez les taux d'accès au cache, le volume des demandes d'origine et la latence de réponse pour le contenu et les segments publicitaires.

Mesures relatives à l'expérience des spectateurs

Suivez les événements de rebuffering, les heures de démarrage et l'engagement des spectateurs, en particulier lors des transitions publicitaires.

Pour des informations détaillées sur la surveillance, reportez-vous [Surveillez les opérations pour le CDN et MediaTailor les intégrations](#) aux sections et [Surveillez les MediaTailor opérations d'assemblage des canaux CDN](#).

Dépannage des problèmes courants

Lorsque vous résolvez des problèmes liés à la solution intégrée, tenez compte des problèmes courants suivants :

Problèmes de synchronisation des pauses publicitaires

Si les annonces ne s'affichent pas aux points de rupture attendus, vérifiez que les définitions des interruptions publicitaires dans vos programmes d'assemblage de chaînes sont correctement configurées et que l'insertion des publicités identifie correctement ces points de rupture.

Erreurs de livraison manifestes

Si les spectateurs rencontrent des problèmes de lecture, vérifiez que le CDN transmet correctement les demandes de manifeste entre l'assemblage des chaînes et l'insertion des publicités, et que les paramètres de cache sont adaptés à la nature dynamique de ces manifestes.

Problèmes de routage des segments

Si le contenu ou les segments publicitaires ne se chargent pas, vérifiez que le CDN achemine correctement les demandes de segment vers les origines appropriées et que le segment URLs des manifestes est correctement formaté.

Dégradation des performances

Si les utilisateurs sont confrontés à une mise en mémoire tampon ou à une latence élevée, vérifiez les taux de réussite du cache du CDN et le volume des demandes d'origine afin d'identifier les éventuels goulots d'étranglement dans le pipeline de diffusion.

Pour plus d'informations sur le dépannage, consultez [Résoudre les problèmes liés à MediaTailor SSAI CDNs pour une diffusion ininterrompue des publicités](#).

Bonnes pratiques

Suivez ces bonnes pratiques pour une intégration réussie du SSAI à l'assemblage des canaux :

- Testez de manière approfondie : testez la solution intégrée avec différents types de contenu, scénarios publicitaires et conditions d'audience avant de la déployer en production.
- Surveillance continue : configurez une surveillance et des alertes complètes pour identifier et résoudre rapidement les problèmes éventuels.
- Optimisez la mise en cache : passez régulièrement en revue et ajustez vos paramètres de mise en cache du CDN en fonction des modèles d'utilisation réels et des indicateurs de performance.
- Planifiez l'évolutivité : concevez votre architecture de manière à gérer les pics de trafic, en particulier pour les chaînes ou les événements populaires.
- Envisagez la redondance : implémentez la redondance dans les composants critiques afin de garantir la haute disponibilité de vos canaux linéaires.
- Optimisez les transitions entre les annonces : assurez des transitions fluides entre le contenu et les publicités en utilisant des profils de codage et des durées de segment cohérents.

Informations connexes

Pour plus d'informations sur l'intégration du SSAI à l'assemblage des canaux, voir :

Documentation d'assemblage des canaux

[Utilisation AWS Elemental MediaTailor pour créer des flux linéaires assemblés](#)- Découvrez les concepts d'assemblage de canaux

[Assemblage de canaux avec CDN](#)- Configurer l'assemblage des canaux avec un CDN

Documentation SSAI

[Insertion d'annonces avec CDN](#)- Configurer l'insertion d'annonces avec un CDN

[Comprendre l'architecture d'insertion publicitaire pour le CDN et les intégrations MediaTailor](#)- Comprendre l'architecture du CDN d'insertion publicitaire

Configuration du CDN

[Configurer l'intégration du CDN](#)- Conseils généraux de configuration du CDN

[intégration d'CloudFront](#)- configuration CloudFront spécifique

Optimisez les performances du CDN et MediaTailor des intégrations

Optimisez les performances de votre mise en œuvre d'insertion d' AWS Elemental MediaTailor annonces en optimisant la configuration de votre réseau de diffusion de contenu (CDN). Ces paramètres garantissent une diffusion de contenu efficace et une expérience utilisateur optimale.

Pour obtenir des conseils détaillés sur la mise en cache et l'optimisation du routage spécifiques aux implémentations de SSAI, consultez. [Optimisation des performances du CDN](#) Le guide d'optimisation consolidé fournit des paramètres de mise en cache complets, une configuration du routage des demandes et des tests de performance qui s'appliquent à toutes les intégrations de MediaTailor CDN.

Défis de performance courants

Les implémentations de SSAI avec CDNs peuvent se heurter à plusieurs problèmes de performance :

Frais généraux de manipulation des manifestes

MediaTailor effectue une manipulation du manifeste en temps réel, ce qui peut introduire de la latence si elle n'est pas optimisée correctement. Les éléments suivants peuvent introduire de la latence :

- Temps de traitement des demandes du serveur de décision publicitaire (ADS)
- Temps nécessaire pour modifier les manifestes contenant des références à des segments publicitaires
- Traitement supplémentaire pour la personnalisation

Problèmes d'efficacité du cache

Les manifestes personnalisés peuvent réduire l'efficacité du cache du CDN pour les raisons suivantes :

- Chaque téléspectateur peut recevoir un manifeste unique
- Les paramètres de session peuvent fragmenter le cache
- Le contenu dynamique nécessite une configuration soignée du cache

Pics de charge d'origine

Une mise en cache incorrecte peut entraîner des pics de charge d'origine pendant :

- Événements à fort trafic
- Actualisations du cache
- Changements de configuration du CDN

Problèmes de lecture liés à la publicité

L'insertion de publicités peut perturber la lecture, par exemple :

- Mise en mémoire tampon pendant les transitions publicitaires
- Différences de qualité entre le contenu et les publicités
- Échec de lecture lorsque les publicités ne peuvent pas être récupérées

Pour obtenir des conseils complets sur l'optimisation des performances, notamment les stratégies de mise en cache, le routage des demandes, les tests de performance et les techniques d'optimisation avancées, voir [Optimisation des performances du CDN](#). Le guide d'optimisation consolidé fournit des paramètres détaillés et des benchmarks qui s'appliquent à toutes les intégrations MediaTailor CDN, y compris les implémentations SSAI.

Résumé des meilleures pratiques

Suivez ces bonnes pratiques pour garantir des performances SSAI optimales avec CDN :

Bonnes pratiques en matière d'architecture

- Choisissez le modèle architectural adapté à votre échelle et à vos exigences
- Déployez des services à proximité pour minimiser le temps de latence
- Mettre en œuvre la redondance pour les composants critiques

Bonnes pratiques en matière de mise en cache

- Utiliser différentes stratégies de mise en cache pour différents types de contenu
- Optimisez les clés de cache pour équilibrer personnalisation et efficacité
- Définissez les paramètres appropriés TTLs en fonction du type de contenu et de la fréquence de mise à jour. Pour des recommandations TTL détaillées, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#).

Bonnes pratiques en matière de diffusion publicitaire

- Optimisez les interactions ADS avec des délais d'expiration et des solutions de secours
- Préparez les publicités en fonction des spécifications du contenu
- Mettez en œuvre une diffusion efficace des segments publicitaires

Surveillance des meilleures pratiques

- Surveillez tous les composants de votre mise en œuvre du SSAI
- Configurer des alertes en cas de dégradation des performances
- Vérifiez et optimisez régulièrement votre configuration

Conseils d'optimisation complets

Pour obtenir des conseils complets sur l'optimisation du CDN, notamment des stratégies de mise en cache détaillées, la configuration du routage des demandes, des tests de performance et des techniques d'optimisation avancées, consultez. [Optimisation des performances du CDN](#) Le guide d'optimisation consolidé fournit des paramètres complets et des benchmarks qui s'appliquent à toutes les intégrations MediaTailor CDN, y compris les implémentations SSAI.

Surveillez les opérations pour le CDN et MediaTailor les intégrations

AWS Elemental MediaTailor fournit des fonctionnalités d'analyse robustes qui, combinées aux indicateurs du réseau de diffusion de contenu (CDN), offrent des informations complètes sur la mise en œuvre de votre SSAI. La présente rubrique couvre :

Pour obtenir des conseils complets sur la surveillance du CDN, y compris les indicateurs essentiels, la configuration des outils de surveillance, la configuration des alertes et les stratégies de dépannage applicables à toutes les MediaTailor implémentations, consultez [Surveillance du CDN](#). Cette rubrique se concentre sur les exigences de surveillance spécifiques à la SSAI et sur les analyses d'insertion de publicités.

- Stratégies de surveillance pour l'intégration du SSAI et du CDN
- Outils d'analyse et méthodes de collecte de données
- Techniques d'optimisation basées sur les données

Indicateurs clés pour le SSAI avec CDNs

Pour suivre efficacement la mise en œuvre de votre SSAI avec CDNs, suivez ces indicateurs essentiels :

Mesures d'insertion de publicités

Taux de remplissage des annonces : pourcentage d'opportunités publicitaires qui ont été saisies avec succès par des annonces.

Taux d'erreur publicitaire : pourcentage de demandes d'annonces ayant entraîné des erreurs.

Temps de réponse aux annonces : temps nécessaire au serveur de décision publicitaire pour répondre aux demandes publicitaires.

Précision de la durée des annonces : dans quelle mesure la durée réelle des annonces insérées correspond à la durée attendue.

Mesures relatives à l'expérience des spectateurs

Taux de mise en mémoire tampon : pourcentage du temps de visionnage consacré à la mise en mémoire tampon.

Temps de démarrage : temps nécessaire pour que la lecture vidéo commence.

Fluidité de la transition entre les publicités : la fluidité avec laquelle le lecteur passe du contenu aux publicités.

Durée de la session : durée pendant laquelle les spectateurs regardent le stream avant d'abandonner le stream.

Outils d'analyse et intégration

Combinez ces outils pour créer une solution d'analyse complète pour votre mise en œuvre du SSAI :

AWS Elemental MediaTailor métriques côté serveur

MediaTailor fournit des statistiques intégrées via Amazon CloudWatch qui permettent de suivre les demandes publicitaires, les réponses et les erreurs. Ces statistiques peuvent être consultées dans la CloudWatch console ou intégrées dans des tableaux de bord personnalisés.

Les MediaTailor indicateurs clés incluent :

- `AdDecisionServer.Ads`: le nombre d'annonces renvoyées par le serveur de décision publicitaire.
- `AdDecisionServer.Duration`: durée totale des annonces renvoyées par le serveur de décision publicitaire.
- `AdDecisionServer.Errors`: le nombre d'erreurs renvoyées par le serveur de décision publicitaire.
- `AdDecisionServer.Latency`: temps de réponse du serveur de décision publicitaire.

Pour une liste complète des MediaTailor statistiques, consultez la section [Surveillance MediaTailor avec Amazon CloudWatch](#).

Analyses CDN

Les fournisseurs de CDN proposent des analyses détaillées des performances de diffusion de contenu. Pour Amazon CloudFront, utilisez CloudWatch les métriques et les journaux CloudFront d'accès Amazon pour analyser les modèles de livraison.

Indicateurs CDN importants à surveiller :

- Nombre de demandes par type de contenu (manifestes ou segments)
- Taux de réussite du cache pour différents types de contenu
- Répartition géographique des spectateurs

- Taux d'erreur par code d'erreur

Suivi côté client

Mettez en œuvre le suivi côté client pour recueillir des indicateurs d'expérience utilisateur qui ne sont pas disponibles côté serveur :

- Événements relatifs aux joueurs (play, pause, search, buffer)
- Taux d'achèvement des affichages publicitaires
- Indicateurs de qualité de service (changements de résolution, débit)
- Modèles d'engagement des spectateurs

Envisagez d'utiliser le suivi MediaTailor côté client pour collecter et communiquer ces indicateurs.

Tableaux de bord intégrés

Créez des tableaux de bord complets qui combinent des indicateurs provenant de plusieurs sources :

- Utilisez des CloudWatch tableaux de bord pour combiner MediaTailor et mesurer CloudFront
- Envisagez des plateformes d'analyse tierces pour une visualisation plus avancée
- Configurez une corrélation entre services pour identifier les relations entre les métriques

Mettre en œuvre une stratégie de surveillance

Suivez ces étapes pour mettre en œuvre une stratégie de surveillance complète pour votre SSAI avec mise en œuvre du CDN :

1. Configuration de la surveillance de base

- Activez CloudWatch les métriques pour MediaTailor
- Configurer la journalisation du CDN et la collecte de métriques
- Implémentez le suivi côté client dans votre lecteur vidéo

2. Création de tableaux de bord personnalisés

- Créez un CloudWatch tableau de bord qui combine les indicateurs clés
- Incluez des visualisations relatives au taux de remplissage des publicités, aux performances du CDN et à l'expérience des spectateurs
- Ajouter des annotations pour les événements importants (modifications de configuration, diffusions majeures)

3. Configuration des alertes

- Configurez des CloudWatch alarmes pour les mesures critiques
- Créez des alarmes composites qui se déclenchent dans plusieurs conditions connexes
- Configuration des canaux de notification (e-mail, SMS, Amazon SNS)

4. Mettre en œuvre des réponses automatisées

- Utilisez CloudWatch les événements pour déclencher des réponses automatisées aux problèmes courants
- Créez des runbooks pour une intervention manuelle en cas de besoin
- Documenter les procédures de dépannage pour différents scénarios d'alerte

Exemple Création d'un tableau de bord complet de surveillance SSAI

Cet exemple montre comment créer un CloudWatch tableau de bord qui combine MediaTailor des CloudFront indicateurs :

```
{
  "widgets": [
    {
      "type": "metric",
      "properties": {
        "metrics": [
          [ "AWS/MediaTailor", "AdDecisionServer.Ads", "Configuration", "your-config-
name" ],
          [ ".", "AdDecisionServer.Errors", ".", "." ]
        ],
        "period": 300,
        "stat": "Sum",
        "region": "us-west-2",
        "title": "Ad Decision Server Performance"
      }
    },
    {
      "type": "metric",
      "properties": {
        "metrics": [
          [ "AWS/CloudFront", "Requests", "DistributionId", "your-distribution-id" ],
          [ ".", "4xxErrorRate", ".", "." ],
          [ ".", "5xxErrorRate", ".", "." ]
        ],
        "period": 300,
        "stat": "Average",
```

```
    "region": "us-east-1",  
    "title": "CDN Performance"  
  }  
}  
]  
}
```

Optimisation axée sur les données

Utilisez les données d'analyse que vous collectez pour optimiser la mise en œuvre de votre SSAI :

Optimisation du cache CDN

Analysez les taux de réussite du cache pour identifier les opportunités d'amélioration :

- Ajustez les paramètres TTL en fonction du type de contenu et de la fréquence de mise à jour. Pour des recommandations TTL détaillées, voir [Étape 1 : configurer la mise en cache du CDN pour une diffusion publicitaire optimale](#).
- Optimisation des paramètres clés du cache pour améliorer l'efficacité du cache
- Envisagez d'implémenter Origin Shield pour la mise en cache multicouche

Optimisation de la diffusion des publicités

Utilisez les indicateurs de performance des publicités pour améliorer la diffusion des annonces :

- Identifiez et corrigez les erreurs courantes d'insertion d'annonces
- Optimisez les temps de réponse des serveurs de décision publicitaire
- Ajustez les paramètres de ciblage publicitaire en fonction de l'analyse du taux de remplissage

Optimisation de l'expérience utilisateur

Améliorez l'expérience des spectateurs en fonction des indicateurs côté client :

- Analysez les modèles de baisse pendant les pauses publicitaires
- Optimisez les points de transition des annonces pour une lecture plus fluide
- Ajustez la fréquence et la durée des publicités en fonction des données d'engagement des spectateurs

Optimisation des coûts

Équilibrez les considérations relatives aux performances et aux coûts :

- Analysez les modèles d'utilisation de la bande passante pour optimiser les coûts du CDN
- Envisagez des ajustements de classe de prix pour les CloudFront distributions
- Évaluez le rapport coût-avantage des différentes stratégies de mise en cache

Bonnes pratiques

Suivez les meilleures pratiques suivantes pour une surveillance et une analyse efficaces du SSAI :

- Établir des bases de référence : collectez des métriques pendant le fonctionnement normal afin d'établir des bases de performance qui peuvent être utilisées à des fins de comparaison lors du dépannage.
- Mettez en œuvre une surveillance à plusieurs niveaux : surveillez à différents niveaux de votre architecture (origine, CDN, lecteur) pour obtenir une image complète des performances.
- Corrélisez les indicateurs entre les services : recherchez les relations entre les indicateurs des différents services afin d'identifier les causes profondes des problèmes.
- Utilisez la détection des anomalies : mettez en œuvre la détection des CloudWatch anomalies pour identifier automatiquement les modèles inhabituels dans vos indicateurs.
- Révissez et affinez régulièrement : planifiez des révisions régulières de votre stratégie de surveillance et ajustez-la en fonction de l'évolution des exigences et des nouvelles informations.
- Documenter les résultats et les actions : Tenir un registre des efforts d'optimisation et de leurs résultats pour renforcer les connaissances institutionnelles.

Informations connexes

Pour plus d'informations sur la surveillance et l'analyse du SSAI avec CDNs, voir :

- [Optimisez les performances du CDN et MediaTailor des intégrations](#) pour les techniques d'optimisation des performances
- [Résoudre les problèmes liés à MediaTailor SSAI CDNs pour une diffusion ininterrompue des publicités](#) pour résoudre les problèmes courants
- [Surveillance MediaTailor avec Amazon CloudWatch](#) pour obtenir des informations détaillées sur MediaTailor les métriques
- [Affichage CloudFront des métriques et des fonctions Edge](#) pour obtenir des informations sur CloudFront les métriques

Résoudre les problèmes liés à MediaTailor SSAI CDNs pour une diffusion ininterrompue des publicités

Cette section fournit des solutions aux problèmes courants liés à l'utilisation de l'insertion AWS Elemental MediaTailor dynamique d'annonces avec un réseau de diffusion de contenu (CDN). Ces solutions vous aideront à résoudre les problèmes liés à la monétisation de vos vidéos grâce à des publicités personnalisées.

Pour obtenir des conseils complets sur le dépannage du CDN, notamment sur les problèmes de performances du cache, la résolution des erreurs HTTP, les procédures de test et les techniques de diagnostic applicables à toutes les MediaTailor implémentations, consultez. [Résoudre les problèmes liés à l'intégration du CDN](#) Cette section se concentre sur les exigences de dépannage spécifiques à la SSAI et les problèmes d'insertion de publicités.

Si vous rencontrez des problèmes avec la configuration de votre CDN et de votre SSAI, vérifiez les problèmes courants suivants :

La publicité personnalisée n'apparaît pas dans le stream

Vérifiez que votre ADS répond correctement et qu'il AWS Elemental MediaTailor communique avec lui. Vérifiez les problèmes potentiels suivants :

- Les paramètres de requête de ciblage publicitaire ne sont pas correctement transmis via votre CDN
- Les points de rupture des publicités sont mal définis dans votre contenu
- Problèmes de connectivité ou de réponse ADS

Erreurs de lecture aux points de rupture des publicités

Assurez-vous que les segments publicitaires sont correctement transcodés pour correspondre aux débits et résolutions de votre contenu. Vérifiez les problèmes courants suivants :

- Le CDN achemine incorrectement les demandes de segments publicitaires
- Erreurs de manipulation manifestes aux points de transition
- Profils d'encodage incompatibles entre le contenu et les publicités

Manifestes périmés

Pour le contenu en direct, vérifiez que les paramètres TTL de votre cache CDN sont appropriés. Pour les manifestes personnalisés, utilisez un TTL de 0 seconde. Envisagez de mettre en œuvre

l'invalidation du cache pour les manifestes qui changent rapidement. Pour des conseils complets sur le TTL, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#).

Latence élevée

Vérifiez la configuration de votre CDN pour un routage optimal. Assurez-vous que votre CDN dispose d'emplacements périphériques proches de vos spectateurs pour de meilleures performances.

Dépannage avancé

Pour les problèmes plus complexes, essayez les techniques de résolution avancées suivantes :

Comportement de ciblage publicitaire incohérent

Vérifiez les incohérences des paramètres de requête entre les demandes des joueurs et les demandes ADS. Assurez-vous que tous les paramètres de ciblage requis sont correctement transmis via le CDN.

Incohérences du cache du CDN

Vérifiez les configurations des clés de cache pour garantir une différenciation correcte du contenu. Envisagez d'implémenter la purge du cache pour les mises à jour critiques des manifestes.

Défaillances du suivi des publicités

Vérifiez que les balises URLs sont correctement transmises et qu'elles ne sont pas bloquées par le CDN. Vérifiez que les joueurs clients peuvent accéder aux points de suivi.

Optimisation des performances

Pour optimiser les performances de votre flux de travail dynamique d'insertion publicitaire et de monétisation vidéo :

- Ajustez les paramètres TTL en fonction du type de contenu et des modèles de visionnage. Pour des recommandations TTL détaillées, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#).
- Mettez en œuvre le routage géographique afin de minimiser le temps de latence pour le public mondial
- Envisagez d'en utiliser plusieurs CDNs pour une redondance et des performances optimales

- Surveillez les taux d'accès au cache et ajustez les configurations en conséquence
- Optimisez les processus de manipulation des manifestes afin de réduire le temps de traitement aux points de rupture des publicités
- Pré-transcodez les publicités pour qu'elles correspondent aux profils de contenu courants pour une insertion publicitaire personnalisée et fluide

Critères de performance

Lorsque vous optimisez l'intégration de votre CDN d'insertion d' AWS Elemental MediaTailor annonces, visez les critères de performance suivants :

Objectifs en matière de taux de réussite du cache

Segments de contenu : taux d'accès au cache supérieur à 95 %

Segments publicitaires : taux de réussite du cache supérieur à 90 %

Manifestes : Non applicable (ne doivent pas être mis en cache pour l'insertion d'annonces personnalisées)

Repères de latence

Latence des demandes de manifeste : moins de 100 ms (P95)

Diffusion du segment de contenu : moins de 50 ms (P95)

Diffusion du segment publicitaire : moins de 75 ms (P95)

End-to-end temps de démarrage : moins de 2 secondes

Métriques de charge d'origine

Demandes d'origine par téléspectateur : moins de 0,1 demande par minute et par téléspectateur

Bande passante d'origine par spectateur : moins de 5 % de la bande passante totale du spectateur

Objectifs en matière de taux d'erreur

Erreurs manifestes : moins de 0,1 %

Erreurs de segment : moins de 0,01 %

Rebuffering déclaré par le joueur : moins de 1 %

Benchmarks d'évolutivité

Support d'un trafic 10 fois supérieur à la normale pendant les périodes de pointe, sans dégradation

Capacité à traiter plus de 1 000 demandes par seconde et par canal

Utilisez CloudWatch les métriques Amazon pour suivre ces indicateurs de performance. Pour des instructions de surveillance détaillées, voir [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#).

Informations connexes

Pour plus d'informations sur l'insertion d'annonces avec CDNs, voir :

Documentation d'insertion d'annonces

[Commencer à insérer MediaTailor des publicités](#)- En savoir plus sur les concepts d'insertion de publicités

[Configuration](#)- Commencez par insérer des publicités

Intégration au CDN

[Configurer l'intégration du CDN](#)- Conseils généraux de configuration du CDN

[intégration d'CloudFront](#)- configuration CloudFront spécifique

Intégration de l'assemblage des canaux

[Assemblage de canaux avec CDN](#)- En savoir plus sur l'assemblage des canaux avec CDNs

[Mettre en œuvre l'insertion de publicités](#)- Implémenter l'insertion de publicités avec l'assemblage des chaînes

Surveillance et optimisation

[Surveillez les opérations pour le CDN et MediaTailor les intégrations](#)- Surveillance et analyse complètes

[Optimisez les performances du CDN et MediaTailor des intégrations](#)- Guide d'optimisation des performances

[Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#)- CloudWatch métriques pour MediaTailor

Créez des canaux MediaTailor linéaires avec assemblage de canaux et CDN

Cette section fournit des conseils complets pour intégrer l'assemblage de AWS Elemental MediaTailor canaux à un réseau de diffusion de contenu (CDN). Suivez ces étapes pour configurer, configurer et optimiser l'intégration de votre CDN d'assemblage de canaux.

Vous pouvez également combiner l'assemblage des chaînes avec l'insertion de publicités côté serveur (SSAI) pour créer des chaînes linéaires monétisées avec une publicité personnalisée. Cette puissante intégration vous permet de diffuser des publicités ciblées à différents spectateurs qui regardent le contenu d'une même chaîne, augmentant ainsi vos opportunités de revenus tout en garantissant une expérience de visionnage de qualité. Pour plus d'informations sur le SSAI avec CDNs, voir [Insertion d'annonces avec CDN](#).

Dans cette rubrique, nous utilisons le terme manifestes pour désigner collectivement les listes de lecture multivariantes, les listes de lecture multimédia et MPDs.

Pour plus d'informations sur l'assemblage des MediaTailor canaux, consultez [Utilisation AWS Elemental MediaTailor pour créer des flux linéaires assemblés](#).

Ce dont vous aurez besoin

Avant de configurer l'assemblage de MediaTailor canaux avec un CDN, rassemblez les ressources requises suivantes :

AWS compte et autorisations

Un AWS compte doté des autorisations appropriées pour créer et gérer MediaTailor des ressources

Autorisations IAM pour MediaTailor, CloudFront (si vous en utilisez), et les services connexes

Pour connaître les exigences détaillées en matière d'autorisation, voir [Sécurité dans AWS Elemental MediaTailor](#).

Services requis

Un MediaTailor canal d'assemblage de canaux en cours d'exécution (pas simplement une configuration SSAI)

Un compte de réseau de diffusion de contenu (CDN) (Amazon CloudFront ou CDN tiers)

Stockage d'origine pour votre contenu VOD (Amazon S3 ou autre serveur d'origine)

MediaPackage

Exigences relatives au contenu

Les sources VOD sont correctement encodées et emballées au format HLS ou DASH. Pour plus d'informations sur l'utilisation des emplacements de source et des sources de VOD, consultez [Utilisation des emplacements sources](#).

Contenu avec des durées de segment cohérentes (minimum recommandé : 1 seconde)

Contenu de la liste d'annonces pour les pauses publicitaires (si vous implémentez l'insertion d'annonces). Pour plus d'informations sur la configuration de Slate, consultez [MediaTailor ardoise et insertion](#).

Avant de commencer

Important

Ce flux de travail nécessite un canal d'assemblage de MediaTailor canaux en cours d'exécution. Le fait de n'avoir qu'une configuration MediaTailor SSAI n'est pas suffisant pour cette intégration. Un canal d'assemblage de canaux actif doit être configuré et opérationnel avant de procéder à l'intégration du CDN.

Effectuez les tâches de configuration suivantes avant de mettre en œuvre l'assemblage de MediaTailor canaux avec un CDN :

1. Configurer la connectivité réseau entre votre CDN et MediaTailor les serveurs d'origine
2. Configurer le protocole HTTPS pour une diffusion de contenu sécurisée
3. Configurer les paramètres DNS pour votre domaine CDN

Connaissances requises

Pour implémenter cette solution avec succès, vous devez disposer des éléments suivants :

- Compréhension des protocoles de streaming (HLS/DASH)

- Connaissances de base des principes de configuration du CDN
- Connaissance des concepts d'assemblage de MediaTailor canaux

Pour la MediaTailor configuration de base, reportez-vous à [Configuration](#) et [Débuter avec l'assemblage de MediaTailor canaux](#). Pour plus d'informations sur l'utilisation des emplacements de source et des sources de VOD, consultez [Utilisation des emplacements sources](#). Pour plus d'informations sur la configuration de Slate, consultez [MediaTailor ardoise et insertion](#).

Avantages de l'intégration du CDN

L'intégration de l'assemblage de canaux à un CDN offre ces principaux avantages.

Expérience utilisateur améliorée

Une configuration correcte CDNs réduit la mise en mémoire tampon, les temps de démarrage et les erreurs de lecture pour les canaux linéaires. Cela se traduit par un engagement et une satisfaction accrus des spectateurs.

Réduction des coûts

Des stratégies de mise en cache efficaces minimisent les demandes d'origine. Cela réduit les coûts de transfert de données et la charge du serveur d'origine, ce qui est particulièrement important pour les canaux linéaires à haut volume.

Capacité de mise à l'échelle

Les configurations CDN optimisées gèrent les pics de trafic lors d'événements populaires sans dégrader les performances, garantissant ainsi la disponibilité de vos canaux linéaires même pendant les heures de pointe.

Portée mondiale

Configuré correctement, CDNs diffusez du contenu avec une faible latence aux spectateurs du monde entier, où qu'ils se trouvent, élargissant ainsi votre audience potentielle.

Transitions de programme fluides

La configuration optimisée du CDN garantit des transitions fluides entre les programmes de votre chaîne linéaire, créant ainsi une expérience de visionnage de qualité télévisuelle.

Rubriques

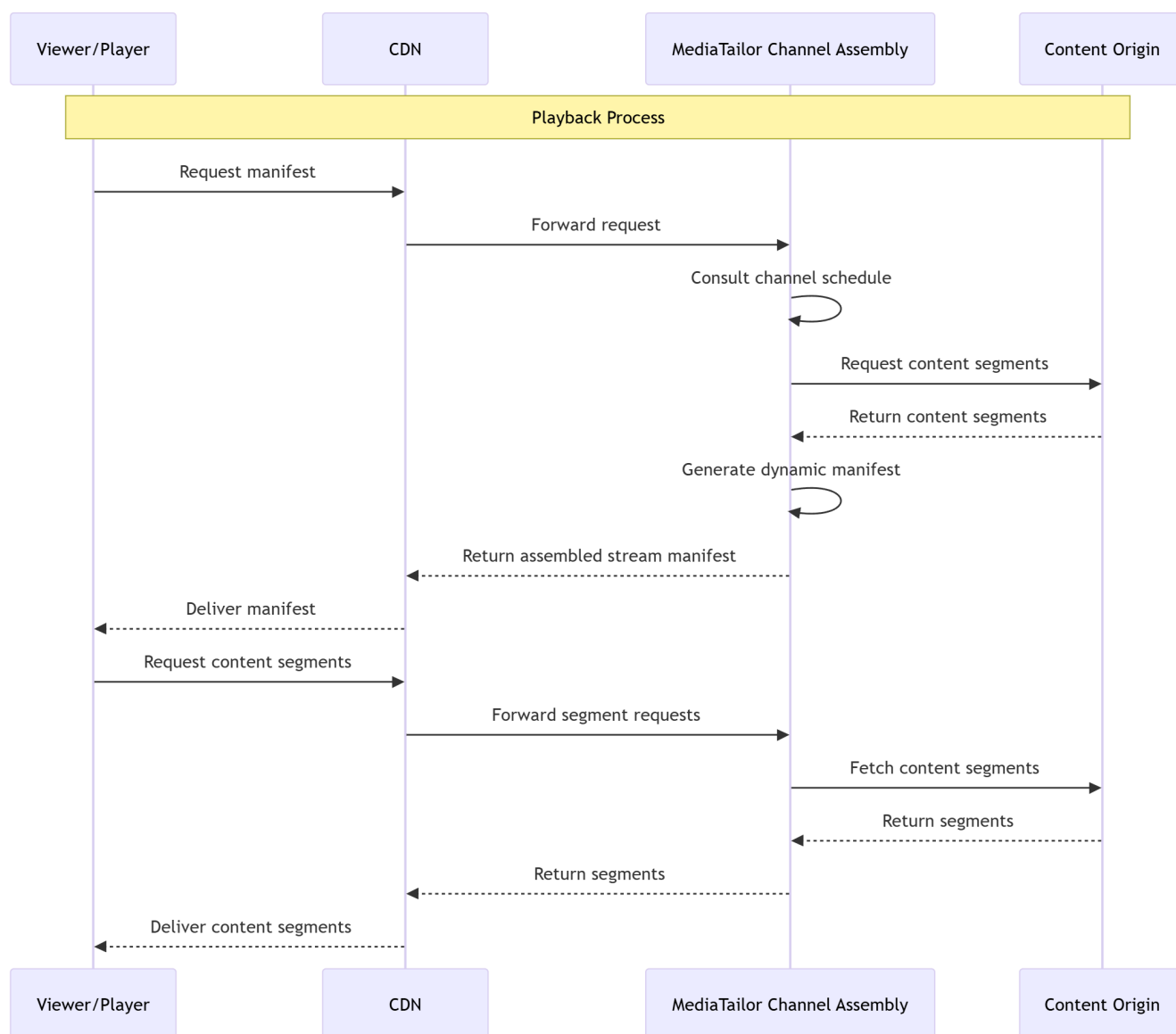
- [Comprendre l'architecture CDN d'assemblage de MediaTailor canaux](#)
- [Configurer l'assemblage de MediaTailor canaux de base avec un CDN](#)
- [Configurer la base URLs pour le CDN d'assemblage de MediaTailor canaux](#)
- [MediaTailor Implémenter l'insertion de publicités avec l'assemblage de chaînes](#)
- [Configuration de l'affichage décalé dans le temps pour l'assemblage des MediaTailor canaux](#)
- [Surveillez les MediaTailor opérations d'assemblage des canaux CDN](#)
- [Optimisation complète de l'assemblage des canaux \(CDN\)](#)

Comprendre l'architecture CDN d'assemblage de MediaTailor canaux

AWS Elemental MediaTailor l'assemblage des canaux s'intègre aux réseaux de diffusion de contenu (CDNs) pour diffuser des chaînes de streaming linéaires avec des performances améliorées et une portée mondiale. L'architecture recommandée place le CDN entre les spectateurs et l'assemblage des chaînes, l'assemblage des chaînes accédant au contenu directement depuis votre origine. Cette rubrique décrit les principaux composants de l'architecture et explique comment ils fonctionnent ensemble pour diffuser votre contenu.

1. Les spectateurs demandent des manifestes au CDN
2. Le CDN transmet les demandes à l'assemblage des canaux
3. L'assemblage des canaux assemble les manifestes à partir de sources VOD
4. L'assemblage des chaînes renvoie les manifestes au CDN, qui les transmet aux spectateurs
5. Les spectateurs demandent des segments via le CDN
6. Le CDN achemine les demandes de segment vers l'origine appropriée

Cette architecture garantit des performances optimales tout en conservant les avantages de sécurité et de flexibilité liés à l'utilisation d'un CDN.



Terminologie CDN pour l'assemblage de canaux

La compréhension de ces termes clés vous aidera à implémenter et à résoudre les problèmes liés à l'intégration de votre CDN d'assemblage de canaux :

CDN d'origine et CDN Edge

CDN d'origine : un CDN positionné entre MediaTailor et l'origine de votre contenu. Il met en cache les segments de contenu afin de réduire la charge sur vos serveurs d'origine. Dans une architecture multi-CDN, il s'agit de la première couche CDN qui s'interface directement avec l'origine.

Edge CDN : un CDN positionné entre les spectateurs et MediaTailor. Il fournit des manifestes et du contenu personnalisés aux spectateurs. Dans une architecture multi-CDN, il s'agit de la couche CDN la plus externe qui s'interface directement avec les spectateurs.

Termes de configuration du CDN

Comportement du cache : règles qui déterminent la manière dont un CDN gère différents types de demandes, notamment la durée de mise en cache et le routage d'origine.

TTL (Time To Live) : durée pendant laquelle le contenu reste valide dans un cache CDN avant de devoir être actualisé depuis son origine. Pour des recommandations TTL détaillées, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#).

Clé de cache : identifiant unique utilisé par un CDN pour stocker et récupérer le contenu mis en cache, notamment le chemin d'URL, les paramètres de requête et les en-têtes.

Origin Shield : couche de mise en cache intermédiaire entre les emplacements périphériques du CDN et votre serveur d'origine qui réduit le nombre de demandes adressées à votre origine.

Réduction des demandes : fonctionnalité CDN qui combine plusieurs demandes simultanées pour le même contenu en une seule demande d'origine.

MediaTailor-termes spécifiques au CDN

Préfixe de segment de contenu CDN : nom de domaine CDN MediaTailor utilisé lors de la génération de segments de contenu dans URLs des manifestes.

Préfixe de segment publicitaire CDN : nom de domaine CDN MediaTailor utilisé lors de la génération de segments publicitaires dans URLs des manifestes.

Pour plus d'informations sur la configuration du CDN avec MediaTailor, consultez [Configurer l'intégration du CDN](#).

Configurer l'assemblage de MediaTailor canaux de base avec un CDN

AWS Elemental MediaTailor L'assemblage des chaînes vous permet de configurer une intégration de base avec votre réseau de diffusion de contenu (CDN) pour une diffusion efficace des chaînes de streaming linéaires à vos spectateurs. Suivez ces étapes pour configurer l'intégration entre l'assemblage des canaux et votre CDN.

1. Configurez votre CDN pour qu'il accepte les manifestes des spectateurs et les transmette à l'assemblage des MediaTailor chaînes.

2. Configurez l'assemblage des MediaTailor chaînes pour accéder au calendrier de vos chaînes et déterminer la programmation en cours.
3. Configurez l'assemblage des MediaTailor chaînes pour demander des segments de contenu à votre serveur d'origine en fonction du calendrier.
4. Assurez-vous que l'origine de votre contenu peut fournir les segments demandés à l'assemblage des MediaTailor chaînes.
5. Configurez l'assemblage des MediaTailor canaux pour générer des manifestes dynamiques en fonction du calendrier actuel.
6. Configurez votre CDN pour diffuser les playlists multivariantes assemblées, les playlists multimédias et les diffuser MPDs aux spectateurs.
7. Configurez votre CDN pour traiter les demandes de segment des utilisateurs, avec les paramètres de cache appropriés.
8. Configurez votre CDN pour transférer les erreurs de cache à l'assemblage des MediaTailor canaux.
9. Configurez l'assemblage des MediaTailor chaînes pour récupérer les segments demandés à partir de l'origine de votre contenu.
10. Configurez votre CDN pour diffuser des segments de contenu aux spectateurs pour qu'ils puissent les lire.

Configurer la base URLs pour le CDN d'assemblage de MediaTailor canaux

AWS Elemental MediaTailor l'assemblage des canaux nécessite une configuration d'URL de base appropriée pour garantir le bon fonctionnement du routage du contenu via votre réseau de diffusion de contenu (CDN). Configurez les paramètres d'URL de base dans l'assemblage des chaînes pour permettre la diffusion réussie du contenu aux spectateurs.

Configuration de l'URL du segment de contenu

Dans la configuration de votre canal d'assemblage de canaux, définissez l'URL de base de votre domaine CDN. Cela garantit que tous les segments URLs du manifeste assemblé pointent vers votre CDN plutôt que directement vers votre serveur d'origine.

Par exemple, si votre contenu d'origine se trouve à `http://origin.example.com/content/` et que votre domaine CDN l'est `https://cdn.example.com/`, définissez l'URL de base sur `https://cdn.example.com/content/`.

Configuration des restrictions d'accès

Pour améliorer la sécurité, configurez votre CDN pour restreindre l'accès direct à votre serveur d'origine :

1. Configurez les contrôles d'accès à l'origine dans votre CDN.
2. Configurez votre serveur d'origine pour qu'il accepte uniquement les demandes provenant de votre CDN.
3. Utilisez des cookies signés URLs ou des cookies pour authentifier les spectateurs si nécessaire.

Pour Amazon CloudFront, vous pouvez utiliser le contrôle d'accès à l'origine (OAC) pour sécuriser l'accès à votre origine. Pour plus d'informations sur la sécurisation de votre CloudFront intégration, consultez [intégration d'CloudFront](#).

MediaTailor Implémenter l'insertion de publicités avec l'assemblage de chaînes

L'assemblage des chaînes AWS Elemental MediaTailor s'intègre parfaitement à l'insertion publicitaire côté serveur (SSAI) et aux réseaux de diffusion de contenu (CDNs) pour créer des canaux linéaires monétisés avec des publicités personnalisées.

Lorsque vous combinez l'assemblage de chaînes avec le SSAI, vous pouvez créer des chaînes linéaires qui diffusent des publicités personnalisées aux spectateurs tout en garantissant des expériences de qualité. Cette intégration vous permet d'effectuer les opérations suivantes :

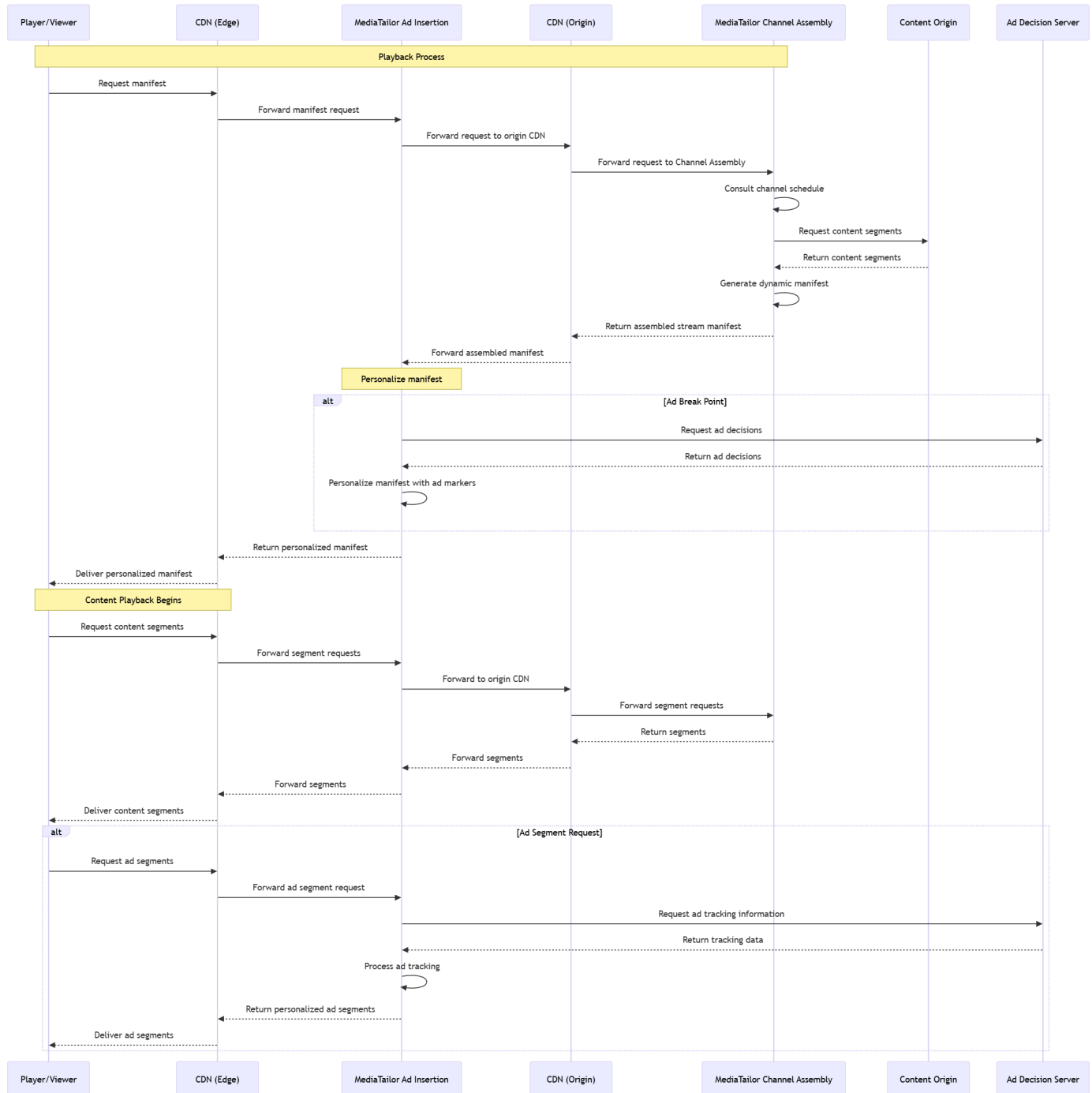
- Monétisez le contenu - Générez des revenus grâce à une publicité ciblée sur vos canaux linéaires
- Personnalisez les expériences : diffusez différentes publicités aux spectateurs qui regardent la même chaîne en fonction de leur profil
- Maintien de la qualité - Garantisiez des transitions fluides entre le contenu et les publicités pour un visionnage de qualité
- Évoluez efficacement - Support de millions de spectateurs simultanés grâce à la diffusion de CDN

Pour des informations détaillées sur le SSAI avec CDNs, voir [Insertion d'annonces avec CDN](#).

1. Configurez votre CDN Edge pour accepter les demandes de manifeste des spectateurs et les transférer vers l'insertion MediaTailor publicitaire.

2. Configurez l'insertion d' MediaTailor annonces pour transférer les demandes vers votre CDN d'origine.
3. Configurez votre CDN d'origine pour transmettre les demandes à l'assemblage des MediaTailor canaux.
4. Configurez l'assemblage des MediaTailor canaux pour générer des manifestes dynamiques en fonction du calendrier actuel.
5. Configurez votre CDN d'origine pour transférer les manifestes assemblés vers l'insertion MediaTailor publicitaire.
6. Configurez l'insertion d' MediaTailor annonces pour demander des décisions publicitaires à votre serveur de décision publicitaire aux points de rupture des publicités.
7. Configurez l'insertion de MediaTailor publicités pour personnaliser les manifestes à l'aide de marqueurs publicitaires.
8. Configurez votre CDN Edge pour fournir des manifestes personnalisés aux spectateurs.
9. Configurez votre architecture CDN pour gérer efficacement les demandes de contenu et de segments publicitaires.

Le schéma suivant illustre ce flux de travail combiné :



Pour des performances optimales lors de la combinaison de l'assemblage des canaux et du SSAI :

- Configurer des comportements de cache qui font la distinction entre l'assemblage de canaux et les demandes SSAI

- Définissez les valeurs TTL appropriées pour les manifestes et les segments, comme recommandé dans [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
- Garantisiez un routage correct entre l'assemblage des chaînes, l'insertion de publicités et les origines de votre CDN
- Surveillez les indicateurs de performance pour les composants d'assemblage des chaînes et d'insertion de publicités

Pour des informations détaillées sur la configuration de SSAI avec CDNs, voir :

- [Comprendre l'architecture d'insertion publicitaire pour le CDN et les intégrations MediaTailor](#)- En savoir plus sur l'architecture et les concepts du SSAI
- [Configurez un MediaTailor SSAI de base avec un CDN pour une diffusion publicitaire optimale](#)- Instructions Step-by-step de configuration SSAI
- [Résoudre les problèmes liés à MediaTailor SSAI CDNs pour une diffusion ininterrompue des publicités](#)- Résoudre les problèmes courants d'intégration SSAI

Configuration de l'affichage décalé dans le temps pour l'assemblage des MediaTailor canaux

AWS Elemental MediaTailor l'assemblage des canaux prend en charge les fonctionnalités de visualisation décalées dans le temps qui permettent des fonctionnalités similaires à celles du DVR, telles que la pause, le retour en arrière et le redémarrage pour vos canaux linéaires. Activez ces fonctionnalités en configurant votre réseau de diffusion de contenu (CDN) pour prendre en charge le visionnage décalé dans le temps, ce qui permet aux spectateurs de contrôler leur expérience de visionnage.

Comprendre le visionnage décalé dans le temps

L'affichage décalé dans le temps permet des fonctionnalités similaires à celles d'un DVR pour les canaux linéaires, notamment :

- Redémarrage : les spectateurs peuvent commencer à regarder une émission depuis le début, même s'ils la rejoignent après le début
- Pause et reprise : les spectateurs peuvent suspendre le contenu et reprendre le visionnage plus tard

- Revenir en arrière et avancer rapidement : les spectateurs peuvent naviguer en arrière et en avant dans le contenu disponible
- Visionnage différé : les spectateurs peuvent regarder le contenu diffusé plus tôt dans le programme de la chaîne

L'affichage décalé dans le temps fonctionne en ajoutant un `start` paramètre à l'URL de diffusion de la chaîne. Le paramètre indique un décalage en secondes par rapport à l'heure actuelle :

- Les valeurs négatives indiquent un moment dans le passé (par exemple, `start=-3600` cela signifie « commencer il y a 1 heure »)
- Les valeurs positives indiquent une heure dans le futur (par exemple, `start=3600` cela signifie « commencer à partir d'une heure dans le futur »)

Exemple d'URL avec paramètre de décalage temporel :

```
https://example-cdn.com/out/v1/channel-name/index.m3u8?start=-3600
```

Résolution du délai

Pour prendre en charge le visionnage décalé dans le temps avec un CDN :

1. Configurez votre CDN pour transmettre le paramètre de `start` requête à l'assemblage des canaux.
2. Configurez des comportements de cache qui incluent le `start` paramètre dans la clé de cache.
3. Pour les manifestes avec des paramètres de décalage temporel, utilisez un TTL court ou aucune mise en cache.

Cela garantit que chaque téléspectateur reçoit le manifeste correspondant à la position temporelle demandée.

Exigences du CDN pour le décalage horaire

Votre CDN doit répondre aux exigences suivantes pour prendre en charge le visionnage décalé dans le temps avec assemblage de canaux :

- Transférez tous les paramètres de requête à l'assemblage des canaux.
- Incluez le `start` paramètre dans la clé de cache.

- Support d'une invalidation correcte du cache pour les manifestes décalés dans le temps.
- Gérez différentes réponses au manifeste en fonction des paramètres de requête.

Surveillez les MediaTailor opérations d'assemblage des canaux CDN

AWS Elemental MediaTailor l'assemblage des canaux nécessite une surveillance efficace lorsqu'il est intégré à un réseau de diffusion de contenu (CDN) afin de garantir une diffusion de contenu fiable. Mettez en œuvre des stratégies de surveillance pour l'assemblage de vos chaînes et l'intégration du CDN afin de garantir une diffusion de contenu fiable et une résolution rapide des problèmes.

Pour obtenir des conseils complets sur la surveillance du CDN, y compris les indicateurs essentiels, la configuration des outils de surveillance, la configuration des alertes et les stratégies de dépannage applicables à toutes les MediaTailor implémentations, consultez [Surveillance du CDN](#). Cette section se concentre sur les exigences de surveillance spécifiques à l'assemblage des canaux.

Mettez en œuvre une surveillance spécifique pour l'assemblage de vos canaux et l'intégration du CDN :

- Suivez les métriques de génération de manifestes lors de l'assemblage des canaux.
- Surveillez les demandes de visionnage décalées dans le temps et leur impact sur les taux de réussite du cache CDN.
- Configurez des alertes pour détecter des modèles inhabituels dans les demandes de manifeste.
- Mettez en œuvre le suivi de la disponibilité des segments dans l'ensemble de vos sources de contenu.

Pour Amazon CloudFront, créez un tableau de bord qui intègre les métriques du CDN aux MediaTailor métriques afin de visualiser l'ensemble de votre pipeline de livraison à l'aide de [CloudWatchtableaux](#) de bord.

Si vous utilisez également le SSAI pour l'assemblage de vos chaînes, consultez [Surveillez les opérations du CDN](#) les recommandations de surveillance supplémentaires spécifiques à l'insertion de publicités.

Optimisation complète de l'assemblage des canaux (CDN)

Pour obtenir des conseils détaillés sur l'optimisation du CDN spécifiques aux implémentations d'assemblages de canaux, y compris les stratégies de mise en cache, la configuration du routage des

demandes, les tests de performance et les techniques d'optimisation avancées, voir. [Optimisation des performances du CDN](#)

Le guide d'optimisation consolidé fournit :

- Paramètres de mise en cache spécifiques à l'assemblage de canaux avec valeurs TTL recommandées
- Optimisation du routage des demandes pour la livraison de manifestes et de segments
- Benchmarks et objectifs de performance pour toutes les MediaTailor implémentations
- Techniques d'optimisation avancées, notamment Origin Shield et compression
- Guidage du flux de travail combiné pour l'assemblage des canaux avec SSAI

Intégration MediaTailor avec un MediaPackage CDN

AWS Elemental MediaTailor s'intègre AWS Elemental MediaPackage pour diffuser des publicités vidéo personnalisées via un réseau de diffusion de contenu (CDN). MediaPackage est un service de packaging et de création just-in-time vidéo qui prépare et protège votre contenu vidéo en vue de sa diffusion sur Internet. Il prend votre contenu vidéo en direct ou à la demande et le regroupe dans des formats de streaming tels que HLS et DASH, le rendant prêt à être visionné sur différents appareils.

Lorsque vous MediaPackage MediaTailor associez un CDN à un CDN, vous créez un flux de diffusion complet qui diffuse des publicités personnalisées à grande échelle. Le CDN distribue votre contenu dans le monde entier, réduisant ainsi la latence et améliorant l'expérience des spectateurs, tout en MediaTailor insérant des publicités ciblées dans vos flux.

Cette rubrique se concentre sur les étapes d'intégration essentielles pour que votre CDN fonctionne ensemble. MediaTailor MediaPackage Pour les options de configuration avancées, le dépannage et les conseils de surveillance, consultez [Étapes suivantes](#).

Comprendre le flux de travail MediaPackage et le flux de travail du CDN

Avant de configurer votre intégration, il est important de comprendre comment MediaPackage et votre CDN fonctionnent ensemble : MediaTailor

1. Préparation du contenu : MediaPackage reçoit votre contenu vidéo en direct ou à la demande et le regroupe dans des formats de streaming (manifestes et segments HLS ou DASH).
2. Insertion de publicités : MediaTailor demande des manifestes MediaPackage, insère des publicités personnalisées et diffuse les manifestes modifiés aux spectateurs.

3. Distribution mondiale : votre CDN met en cache et distribue à la fois les segments de contenu (de MediaPackage) et les segments publicitaires (de MediaTailor) aux spectateurs du monde entier.
4. Lecture par les spectateurs : les lecteurs vidéo demandent des manifestes via le CDN, qui achemine les demandes de manière appropriée entre MediaTailor (pour les manifestes) et MediaPackage (pour les segments de contenu).

Cette architecture présente plusieurs avantages :

- Évolutivité : le CDN gère une charge de visionnage élevée sans affecter vos serveurs d'origine
- Performances : le contenu est diffusé à partir des sites périphériques les plus proches des spectateurs
- Rentabilité : réduction des coûts de bande passante grâce à la mise en cache
- Fiabilité : plusieurs emplacements périphériques assurent la redondance

Conditions préalables

Avant de commencer, assurez-vous que les composants suivants sont configurés :

1. MediaPackage point de terminaison : point de MediaPackage terminaison configuré qui reçoit et emballe votre contenu vidéo. Pour les instructions de configuration, voir [Getting started with MediaPackage](#) dans le guide de MediaPackage l'utilisateur.
2. MediaTailor configuration : MediaTailor configuration qui utilise votre MediaPackage point de terminaison comme origine du contenu. Pour les instructions de configuration, voir [Intégrer une source de contenu pour l'insertion d' MediaTailor annonces](#).
3. Distribution CDN : distribution CDN (telle que CloudFront) configurée pour fonctionner avec le streaming multimédia. Pour les instructions de configuration, voir [Création d'une distribution](#) dans le guide du CloudFront développeur.
4. Serveur de décision publicitaire : serveur de décision publicitaire configuré qui renvoie des réponses VAST ou VMAP pour l'insertion d'annonces.

Étape 1 : configurer les paramètres essentiels du CDN

Une configuration CDN correcte est essentielle pour une MediaPackage intégration réussie. Des paramètres incorrects peuvent entraîner des échecs de lecture, de mauvaises performances du cache et une augmentation des coûts. Sans les bonnes politiques de cache et le transfert des

paramètres de requête, votre CDN risque de ne pas fournir correctement les manifestes ou de contourner complètement la mise en cache, ce qui entraîne une charge de travail élevée sur le serveur d'origine et une dégradation de l'expérience utilisateur.

Configuration des paramètres de base du cache

La configuration de la mise en cache de base est essentielle car elle MediaPackage utilise des en-têtes de contrôle de cache spécifiques pour optimiser la diffusion du contenu. Sans paramètres de cache appropriés, votre CDN risque d'ignorer ces en-têtes, ce qui entraîne des demandes d'origine inutiles et une latence accrue. Procédez comme suit pour garantir un comportement de mise en cache optimal :

Pour configurer une mise en cache de base compatible avec MediaPackage :

1. Ouvrez vos paramètres CloudFront de distribution dans la CloudFront console.
2. Sélectionnez ou créez une politique de cache pour votre MediaPackage origine.
3. Activez l'option « Origin Cache-Control Headers ».
4. Permet de MediaPackage contrôler le comportement de mise en cache via ses en-têtes de contrôle du cache.

Cette configuration de base permet MediaPackage de définir automatiquement des durées de cache appropriées pour différents types de contenu. Pour implémenter une optimisation avancée du cache avec des valeurs TTL spécifiques et un réglage des performances, effectuez d'abord cette configuration de base, puis passez à [Optimisation de la mise en cache du CDN MediaTailor et de la diffusion de contenu MediaPackage](#).

Configuration des paramètres de requête essentiels

La configuration des paramètres de requête est cruciale pour MediaPackage la fonctionnalité. Votre CDN doit transmettre des paramètres de requête spécifiques pour activer des fonctionnalités telles que l'affichage décalé dans le temps et le streaming à faible latence. Des paramètres de requête incorrects peuvent empêcher ces fonctionnalités de fonctionner et je crains de réduire l'efficacité du cache. Pour configurer le transfert des paramètres de requête, procédez comme suit :

Pour vous assurer que votre CDN transmet les paramètres de requête requis à MediaPackage :

1. Dans vos paramètres CloudFront de distribution, sélectionnez ou créez une politique de cache pour les demandes de manifeste.

2. Sous « Paramètres des clés de cache », sélectionnez « Inclure les chaînes de requête spécifiées ».
3. Ajoutez les paramètres de requête essentiels suivants :
 - `start` et `end` - Pour une fonctionnalité de visualisation décalée dans le temps. Ces paramètres sont transmis pour définir des fenêtres de contenu spécifiques MediaPackage pour le visionnage initial et le visionnage de rattrapage.
 - `_HLS_msnet_HLS_part` - Pour prendre en charge les demandes de lecture LL-HLS
 - `m` - Pour capturer la modification `ti/compame` du point de terminaison. MediaPackage les réponses incluent toujours la `?m=###` balise pour capturer l'heure de modification du point de terminaison. Si le contenu est déjà mis en cache avec une valeur différente pour cette balise, CloudFront demande un nouveau manifeste au lieu de diffuser la version mise en cache
 - `aws.manifestfilter` - Pour la fonctionnalité de filtrage des manifestes. Si vous utilisez le filtrage des manifestes, ce paramètre doit être inclus pour configurer la distribution afin de transmettre la chaîne de `aws.manifestfilter` requête à l'origine MediaPackage, ce qui est nécessaire pour que la fonctionnalité de filtrage des manifestes fonctionne
4. N'incluez que les chaînes de requête que MediaPackage utilisent. L'inclusion de chaînes de requête inutiles réduit l'efficacité du cache en créant plusieurs variantes de cache pour le même contenu.

Ces paramètres activent les MediaPackage fonctionnalités de base de votre CDN. Si vous devez implémenter le filtrage du contenu pour différents niveaux d'abonnement ou types d'appareils, effectuez d'abord cette configuration de base des paramètres de requête, puis passez à [Configurer le filtrage des manifestes avec MediaTailor MediaPackage, et le CDN](#).

Pour plus d'informations sur le mode MediaTailor de transmission des paramètres de requête tels que `start` et `end` via MediaPackage pour un affichage décalé dans le temps, voir [MediaTailor gestion des paramètres de requête pour les origines](#) dans [MediaTailor paramètres de requête manifeste](#).

Configurer le délai de réponse pour LL-HLS

La configuration du délai d'expiration est essentielle pour le HLS à faible latence, car le LL-HLS utilise un mécanisme de « blocage des demandes » dans le cadre duquel le CDN attend de nouveaux segments de contenu. Si votre délai d'attente est trop court, les demandes échoueront avant de MediaPackage pouvoir répondre avec de nouveaux segments, ce qui entraînera des interruptions

de lecture et une mauvaise expérience utilisateur. Configurez les délais d'expiration appropriés pour garantir une lecture fluide du format LL-HLS :

Si vous utilisez le HLS à faible latence, configurez les paramètres de délai d'expiration du CDN :

1. Dans les paramètres de votre CDN, recherchez la configuration du délai d'origine.
2. Définissez la valeur du délai de réponse à au moins trois fois la durée de vos pièces.
3. Par exemple, si la durée de vos pièces est de 0,3 seconde, réglez le délai d'attente à au moins 0,9 seconde.

Cela garantit que le CDN attend suffisamment longtemps pour MediaPackage répondre lorsqu'il utilise le mécanisme de blocage des demandes.

Étape 2 : vérifier votre intégration

Il est essentiel de tester votre intégration pour garantir que tous les composants fonctionnent correctement ensemble avant que vos utilisateurs ne rencontrent des problèmes. L'échec de l'intégration peut entraîner une lecture interrompue, des publicités manquantes ou des performances médiocres. Ce processus de vérification vous aide à identifier et à résoudre les problèmes dans un environnement contrôlé.

Après avoir configuré les paramètres de votre CDN, vérifiez que votre intégration fonctionne correctement en testant l'ensemble du flux de travail, de la demande de contenu à l'insertion de publicités.

Étape 2.1 : tester la lecture de base

Les tests de lecture de base vérifient que votre CDN gère correctement les demandes de manifeste et les transmet à MediaTailor. Ce test permet d'identifier les problèmes de configuration liés aux politiques de cache, au transfert des paramètres de requête et à la gestion des manifestes. Pour tester la livraison de manifeste de base, procédez comme suit :

Vérifiez que votre intégration de base fonctionne en demandant un manifeste via votre CDN :

1. Utilisez un navigateur Web ou un curl pour demander une URL de manifeste via votre CDN.
2. Vérifiez que le manifeste se charge correctement et qu'il contient à la fois du contenu et des segments publicitaires.
3. Vérifiez que le segment URLs de contenu du manifeste pointe vers votre domaine CDN.
4. Vérifiez que le segment publicitaire pointe URLs également vers votre domaine CDN.

Si le manifeste se charge correctement et contient les informations attendues URLs, votre intégration de base fonctionne. Pour des méthodologies de test complètes et des procédures de validation avancées, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#). Pour configurer une surveillance complète des performances et de l'état de votre intégration, consultez [Surveillez les performances pour MediaPackage le CDN et les intégrations MediaTailor](#).

Étape 2.2 : tester la lecture vidéo

Les tests de lecture vidéo garantissent le bon fonctionnement de votre intégration complète end-to-end, y compris l'insertion de publicités et la diffusion de contenu via votre CDN. Ce test permet de vérifier que les segments de contenu et les segments publicitaires sont correctement mis en cache et diffusés, et que l'expérience utilisateur répond à vos normes de qualité. Pour tester toutes les fonctionnalités de lecture, procédez comme suit :

Vérifiez que la lecture vidéo fonctionne correctement avec les publicités insérées :

1. Utilisez un lecteur vidéo (tel que Video.js ou HLS.js) pour lire votre contenu via le CDN.
2. Vérifiez que la vidéo s'affiche correctement sans problème de mise en mémoire tampon.
3. Vérifiez que les publicités sont insérées aux heures prévues pendant la diffusion.
4. Vérifiez que le contenu et les segments publicitaires se chargent à partir de votre CDN (et non directement à partir des origines).

Si la lecture fonctionne correctement avec les publicités, votre intégration fonctionne correctement. Pour des méthodologies de test complètes et des procédures de validation avancées, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#). Si vous rencontrez des problèmes de lecture, de mise en mémoire tampon ou d'insertion de publicités, consultez [Résolution des problèmes MediaPackage, CDN et intégrations MediaTailor](#).

Étapes suivantes

Une fois l'intégration de base terminée, vous pouvez implémenter des fonctionnalités et des optimisations avancées :

Optimisation avancée du CDN

Pour une optimisation détaillée du cache, la configuration TTL et le réglage des performances, consultez [Optimisation de la mise en cache du CDN MediaTailor et de la diffusion de contenu MediaPackage](#).

Filtrage des manifestes

Pour implémenter le filtrage de contenu pour les services hiérarchisés, l'optimisation des appareils ou le contrôle d'accès, voir [Configurer le filtrage des manifestes avec MediaTailor MediaPackage, et le CDN](#).

Résolution des problèmes

Si vous rencontrez des problèmes lors de votre intégration, consultez [Résolution des problèmes MediaPackage, CDN et intégrations MediaTailor](#).

Surveillance des performances

Pour configurer une surveillance complète et comprendre les indicateurs de performance clés, voir [Surveillez les performances pour MediaPackage le CDN et les intégrations MediaTailor](#).

Configurer le filtrage des manifestes avec MediaTailor MediaPackage, et le CDN

AWS Elemental MediaTailor utilise le filtrage des manifestes AWS Elemental MediaPackage pour personnaliser les flux audio et vidéo inclus dans les manifestes fournis aux différents spectateurs via un réseau de diffusion de contenu (CDN). Cela est particulièrement utile pour mettre en œuvre des offres de services à plusieurs niveaux, des optimisations spécifiques aux appareils ou des contrôles d'accès au contenu.

Cette rubrique se concentre spécifiquement sur la mise en œuvre des fonctionnalités de filtrage des manifestes. Avant de mettre en œuvre le filtrage des manifestes, vous devez terminer la configuration de base de l'intégration du réseau de diffusion de contenu. Si vous n'avez pas encore configuré votre intégration de base MediaPackage et votre réseau de diffusion de contenu, commencez par [Intégration MediaTailor avec un MediaPackage CDN](#).

Fonctionnalités de filtrage des manifestes

Avant de mettre en œuvre le filtrage des manifestes, découvrez ce que vous pouvez accomplir grâce à cette fonctionnalité :

Fonctionnalités de filtrage de base

Le filtrage des manifestes fournit plusieurs fonctionnalités clés qui vous aident à contrôler la diffusion du contenu :

- Restreindre l'accès des spectateurs au contenu premium (tel que le 4K HEVC)
- Ciblez des types d'appareils spécifiques avec des flux appropriés
- Filtrez le contenu en fonction des fréquences d'échantillonnage audio, des langues ou des codecs vidéo
- Proposez différents niveaux de qualité à différents abonnés

Cas d'utilisation courants

Ces cas d'utilisation montrent comment le filtrage des manifestes peut répondre à des exigences commerciales spécifiques :

Niveaux d'abonnement

Offrez aux abonnés de base des flux en basse résolution tout en offrant aux abonnés premium un accès au contenu 4K

Exemple : le niveau de base est limité à 720p, le niveau Premium passe à 4K

Optimisation de l'appareil

Diffusez automatiquement les flux appropriés en fonction des capacités de l'appareil

Exemple : les appareils mobiles obtiennent des débits inférieurs, les appareils intelligents TVs obtiennent une meilleure qualité

Gestion de la bande passante

Limitez la qualité des flux pendant les périodes de pointe pour gérer les coûts du réseau

Exemple : réduire le débit maximal lors d'événements à fort trafic

Contenu régional

Diffusez différentes langues audio ou variantes de contenu en fonction de la localisation du spectateur

Exemple : filtrer automatiquement les pistes audio en langue locale

Pour plus d'informations sur les concepts de filtrage des manifestes, consultez la section [Filtrage des manifestes](#) dans le guide de AWS Elemental MediaPackage l'utilisateur.

Configurez votre CDN pour le filtrage des manifestes

La configuration du CDN pour le filtrage des manifestes est essentielle car votre CDN doit transmettre le paramètre de `aws.manifestfilter` requête MediaPackage pour que le filtrage fonctionne. Sans un transfert de chaîne de requête approprié, les paramètres du filtre seront supprimés par le CDN, et tous les spectateurs recevront des manifestes non filtrés, quels que soient leur niveau d'abonnement ou les capacités de leur appareil. Cette configuration garantit que votre logique de filtrage atteint MediaPackage et fonctionne comme prévu.

Pour activer le filtrage des manifestes via votre CDN, vous devez configurer le transfert des chaînes de requête :

1. Dans votre CloudFront distribution, créez ou modifiez le comportement du cache pour les demandes de manifeste.
2. Pour la politique de cache, créez une nouvelle politique ou modifiez une politique existante.
3. Sous Paramètres des clés de cache, choisissez « Inclure les chaînes de requête spécifiées ».
4. Ajoutez `aws.manifestfilter` à la liste des chaînes de requête autorisées.
5. Si vous utilisez également d'autres MediaPackage fonctionnalités, ajoutez leurs paramètres de requête :
 - `startet end` - Pour une visualisation décalée dans le temps
 - `time_delay`- Pour la fonctionnalité de temporisation
 - `_HLS_msnet_HLS_part` - Pour LL-HLS

Pour plus d'informations sur la création de distributions, consultez la section [Créer une distribution](#) dans le guide du CloudFront développeur Amazon.

Mettre en œuvre le filtrage côté client

L'implémentation côté client vous permet de définir la manière dont vos lecteurs vidéo et vos applications demandent le contenu filtré. Cette configuration détermine le contenu que chaque spectateur reçoit en fonction de son niveau d'abonnement, des capacités de l'appareil ou d'autres critères. Une mise en œuvre appropriée garantit que les utilisateurs ne reçoivent que le contenu auquel ils devraient avoir accès, tout en maintenant une efficacité optimale du cache du CDN.

Pour implémenter le filtrage des manifestes dans vos lecteurs vidéo et applications, procédez comme suit :

Comment fonctionne le filtrage

Le processus de filtrage fonctionne comme suit :

1. Votre lecteur vidéo ou votre application demande une URL de manifeste qui inclut les paramètres de filtre
2. Le CDN transmet la demande (y compris les paramètres de requête) à MediaTailor
3. MediaTailor transmet les paramètres du filtre MediaPackage lors de la demande du manifeste d'origine
4. MediaPackage applique les filtres et renvoie un manifeste personnalisé contenant uniquement les variantes correspondant à vos critères
5. MediaTailor traite le manifeste filtré pour l'insertion de publicités et le renvoie au lecteur

Format d'URL pour le filtrage

Comprendre le format d'URL correct est essentiel pour une mise en œuvre réussie du filtrage. Un formatage d'URL incorrect entraînera l'ignorance des paramètres de filtrage ou provoquera des erreurs HTTP. La structure de l'URL doit inclure les paramètres de filtre sous forme de chaînes de requête vers MediaPackage lesquelles votre CDN transmet. Procédez comme suit pour implémenter un formatage d'URL approprié :

Pour implémenter le filtrage des manifestes dans vos lecteurs vidéo, procédez comme suit :

1. Modifiez la demande de manifeste de votre joueur URLs pour inclure les paramètres de filtre appropriés.
2. Utilisez le format d'URL suivant avec les paramètres de requête :

```
https://CloudFront-Domain/v1/master/MediaTailor-Config/index.m3u8?  
aws.manifestfilter=video_codec:h264;audio_language:en-US
```

3. Lorsque votre joueur demande cette URL, il MediaTailor transmet ces paramètres à MediaPackage, ce qui donne lieu à un manifeste filtré.

Scénarios de filtrage courants

Utilisez ces exemples pour implémenter des scénarios de filtrage courants :

Diffusion de contenu spécifique à l'appareil

Pour filtrer en fonction des capacités de l'appareil, ajoutez ce paramètre à votre demande de manifeste :

```
aws.manifestfilter=video_codec:h264;audio_sample_rate:0-44100
```

Cet exemple limite le contenu à la vidéo et à l'audio H.264 avec des fréquences d'échantillonnage allant jusqu'à 44,1 kHz, adaptées aux appareils mobiles.

Restriction relative au contenu premium

Pour limiter l'accès au contenu à haut débit, ajoutez ce paramètre à votre demande de manifeste :

```
aws.manifestfilter=video_bitrate:0-9000000
```

Cet exemple limite les débits vidéo à 9 Mbits/s ou moins, ce qui convient aux niveaux d'abonnement de base.

Sélection de la langue

Pour filtrer des langues audio spécifiques, ajoutez ce paramètre à votre demande de manifeste :

```
aws.manifestfilter=audio_language:fr,en-US,de
```

Cet exemple inclut uniquement les pistes audio en français, en anglais américain et en allemand.

Ciblage par résolution

Pour filtrer en fonction de résolutions vidéo spécifiques, ajoutez ce paramètre à votre demande de manifeste :

```
aws.manifestfilter=video_height:240-360,720-1080
```

Cet exemple inclut les flux vidéo dont la hauteur est comprise entre 240 et 360 pixels et 720 à 1080 pixels, à l'exception des résolutions de milieu de gamme.

Filtrage basé sur les codecs

Pour filtrer des codecs vidéo spécifiques, ajoutez ce paramètre à votre demande de manifeste :

```
aws.manifestfilter=video_codec:h264,h265
```

Cet exemple inclut uniquement les flux vidéo H.264 et H.265, à l'exclusion des autres codecs.

Considérations et limites spéciales

Pour éviter les problèmes courants lors de la mise en œuvre du filtrage des manifestes :

Limites techniques

- Pour les manifestes TS, utilisez des groupes de rendu audio pour éviter de supprimer les flux vidéo multiplexés avec des flux audio filtrés
- Dans les manifestes TS et CMAF, le taux d'échantillonnage audio et le débit vidéo ne sont pas facilement visibles dans le manifeste à des fins de vérification
- Les paramètres de demande ajoutés aux playlists ou aux segments multimédias entraîneront une erreur HTTP 400

Conditions d'erreur

- Si le filtrage aboutit à un manifeste vide (aucun flux ne répond aux critères du filtre), une erreur HTTP 400 MediaPackage sera renvoyée
- Des configurations de filtre conflictuelles (filtres de point de terminaison et paramètres de requête) entraînent des erreurs HTTP 404
- Une syntaxe de filtre non valide ou des types de filtres non pris en charge entraînent des erreurs HTTP 400

Considérations sur les performances

- Chaque combinaison de filtres unique crée une entrée de cache distincte, ce qui peut réduire l'efficacité du cache
- Des filtres complexes comportant de nombreux critères peuvent avoir un impact sur les performances de génération de manifestes
- Envisagez d'utiliser des filtres au niveau du point de terminaison pour les scénarios de filtrage statique afin d'améliorer les performances du cache

Testez votre implémentation de filtrage

Il est essentiel de tester votre mise en œuvre du filtrage des manifestes pour garantir que les spectateurs reçoivent le contenu approprié en fonction de leur niveau d'accès et des capacités de leur appareil. L'échec du filtrage peut se traduire par des niveaux de qualité incorrects, des formats non pris en charge ou des contenus auxquels ils ne devraient pas avoir accès. Des tests complets permettent d'identifier et de résoudre ces problèmes avant qu'ils n'affectent vos spectateurs.

Pour vérifier que le filtrage des manifestes fonctionne correctement, procédez comme suit :

1. Demandez des manifestes avec différents paramètres de filtre et vérifiez les résultats
2. Vérifiez que les manifestes filtrés contiennent uniquement les flux attendus
3. Testez les cas extrêmes (résultats vides, filtres non valides) pour garantir une gestion appropriée des erreurs
4. Vérifiez que votre CDN transmet correctement les paramètres du filtre
5. Testez avec différents appareils et lecteurs pour garantir la compatibilité

Pour résoudre les problèmes de filtrage, consultez [Résoudre les problèmes d'intégration du MediaPackage CDN](#).

Si vous rencontrez des erreurs HTTP 400, des manifestes vides ou des paramètres de filtrage qui ne fonctionnent pas comme prévu, consultez des conseils spécifiques de résolution [Résolution des problèmes MediaPackage, CDN et intégrations MediaTailor](#) des problèmes liés au filtrage des manifestes.

Optimisation de la mise en cache du CDN MediaTailor et de la diffusion de contenu MediaPackage

AWS Elemental MediaTailor nécessite une configuration de mise en cache appropriée du réseau de diffusion de contenu (CDN) pour des performances optimales lorsque vous l'utilisez AWS Elemental MediaPackage comme origine de contenu. MediaPackage fournit des en-têtes de contrôle de cache spécifiques qui indiquent à votre réseau de diffusion de contenu la durée pendant laquelle les différents types de contenu doivent être mis en cache. Le respect de ces recommandations garantit une lecture fluide et une diffusion efficace du contenu.

Cette rubrique se concentre spécifiquement sur l'optimisation du comportement de mise en cache afin d'optimiser les performances et de minimiser les coûts. Avant de mettre en œuvre l'optimisation

avancée de la mise en cache, assurez-vous d'avoir terminé la configuration de base de l'intégration du réseau de diffusion de contenu. Si vous n'avez pas encore configuré votre intégration de base, commencez par [Intégration MediaTailor avec un MediaPackage CDN](#).

MediaPackage en-têtes de contrôle du cache

MediaPackage définit des valeurs TTL spécifiques pour différents types de contenu afin d'optimiser le comportement de mise en cache :

Listes de lecture multivariantes (HLS et LL-HLS)

TTL : la moitié de la durée des segments médiatiques

Raison : ces playlists changent à mesure que de nouveaux segments sont disponibles. Elles nécessitent donc des mises à jour fréquentes

Listes de lecture multimédia (HLS standard)

TTL : la moitié de la durée des segments médiatiques

Raison : Comme les playlists multivariantes, elles sont mises à jour au fur et à mesure que le contenu progresse

Listes de lecture multimédia (LL-HLS)

TTL : 1 seconde

Raison : le streaming à faible latence nécessite des mises à jour très fréquentes

Segments multimédias et segments d'initialisation TS

TTL : 1209600 secondes (14 jours)

Raison : les segments multimédias ne changent pas une fois créés, ils peuvent donc être mis en cache pendant de longues périodes

Segments multimédias et segments d'initialisation CMAF

TTL : 1209600 secondes (14 jours)

Raison : comme les segments TS, ils sont immuables une fois créés

Pour obtenir des recommandations TTL complètes pour tous les MediaTailor flux de travail et des stratégies supplémentaires d'optimisation de la mise en cache, voir. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

Configurer les politiques de cache du CDN

Une configuration correcte des politiques de cache est essentielle pour des performances et une rentabilité optimales. Les différents types de contenu (manifestes, segments, fichiers d'initialisation) ont des exigences de mise en cache différentes. L'utilisation de comportements de cache distincts vous permet d'optimiser la mise en cache pour chaque type de contenu, d'améliorer les taux de réussite du cache et de réduire la charge d'origine. Sans politiques de cache appropriées, vous risquez de rencontrer des demandes d'origine inutiles, d'augmenter les coûts et de réduire les performances de lecture.

Pour respecter correctement les en-têtes de MediaPackage contrôle du cache et optimiser la mise en cache :

1. Ouvrez vos paramètres CloudFront de distribution dans la CloudFront console.
2. Créez des comportements de cache distincts pour les différents types de contenu :
 - Demandes de manifeste (*.m3u8, *.mpd)
 - Segments multimédias (*.ts, *.mp4, *.m4s)
 - Segments d'initialisation
3. Pour chaque comportement de cache, créez ou sélectionnez une politique de cache avec les paramètres suivants :
 - Activer l'option « Origin Cache-Control Headers »
 - Définissez la « politique de demande d'origine » pour transférer les en-têtes nécessaires
 - Configurer le transfert de chaîne de requête en fonction du type de contenu

Comportement du cache du manifeste

Pour les demandes de manifeste (*.m3u8, *.mpd) :

- Modèle de chemin : *.m3u8 et *.mpd
- Politique de cache : Honor Origin Cache-Control Headers
- Chaînes de requête : Transférer des paramètres spécifiques (voir [Optimisation du transfert de chaînes de requêtes](#))
- En-têtes : Transférer tous les en-têtes (pour les exigences minimales, voir) [En-têtes requis pour l'intégration du MediaTailor CDN](#)

Comportement du cache des segments multimédias

Pour les segments multimédias (*.ts, *.mp4, *.m4s) :

- Modèle de chemin : *.ts, *.mp4, *.m4s
- Politique de cache : en-têtes de contrôle du cache Honor Origin (TTL de 14 jours)
- Chaînes de requête : aucune (les segments n'utilisent pas les paramètres de requête)
- Compression : activez pour améliorer les performances de diffusion

Optimisation du transfert de chaînes de requêtes

L'optimisation des chaînes de requête est essentielle à l'efficacité du cache, car des paramètres de requête inutiles créent plusieurs variations de cache pour le même contenu. Chaque combinaison unique de paramètres de requête crée une entrée de cache distincte, ce qui réduit le taux de réussite du cache et augmente le nombre de demandes d'origine. En transférant uniquement les chaînes de requête MediaPackage réellement utilisées, vous optimisez l'efficacité du cache tout en conservant toutes les fonctionnalités.

Configurez votre CDN pour transférer uniquement les chaînes de requête MediaPackage utilisées, améliorant ainsi l'efficacité du cache :

Chaînes de requête essentielles

`startet end` - Pour les fenêtres de visualisation décalées dans le temps

`time_delay` - Pour appliquer un délai au contenu du manifeste

`_HLS_msn`, `_HLS_m`, et `_HLS_part` - Pour les demandes de lecture LL-HLS

Chaînes de requête spécifiques aux fonctionnalités

`aws.manifestfilter` - Pour le [filtrage des manifestes](#)

Important

N'incluez aucune autre chaîne de requête dans votre clé de cache. MediaPackage ignore les paramètres non reconnus et leur inclusion réduit l'efficacité du cache en créant des variations inutiles du cache.

Techniques d'optimisation des performances

Ces optimisations sont configurées sur votre CDN (par exemple CloudFront), et non dans ou. MediaPackage MediaTailor Implémentez ces optimisations supplémentaires pour optimiser les performances du cache :

Bouclier Origin

Origin Shield fournit une couche de mise en cache supplémentaire entre les emplacements périphériques et MediaPackage les points de terminaison de votre CDN. Cela permet de réduire le nombre de requêtes qui atteignent vos MediaPackage points de terminaison, ce qui peut améliorer les performances et réduire les coûts, en particulier lors de pics de trafic ou lorsque les taux d'accès au cache sont inférieurs aux niveaux optimaux. Origin Shield est particulièrement utile pour la diffusion en direct lorsque plusieurs sites périphériques peuvent demander le même contenu simultanément.

Activez Origin Shield pour réduire la charge sur vos MediaPackage terminaux :

1. Dans votre CloudFront distribution, activez Origin Shield pour votre MediaPackage origine.
2. Sélectionnez une région du bouclier d'origine proche de votre point de MediaPackage terminaison.
3. Cela crée une couche de mise en cache supplémentaire qui réduit les demandes à MediaPackage.

Configuration de compression

Activez la compression pour les réponses textuelles :

- Activer la compression pour les fichiers manifestes (*.m3u8, *.mpd)
- Ne compressez pas les segments multimédia (déjà compressés)
- Assurez-vous que tous les en-têtes sont transférés vers MediaPackage (pour les exigences minimales, voir [En-têtes requis pour l'intégration du MediaTailor CDN](#))

Surveiller les performances du cache

Suivez ces indicateurs clés pour garantir des performances de cache optimales :

Taux de réussite du cache

Objectif : 90 % ou plus pour les segments médiatiques, 70 % ou plus pour les manifestes

De faibles ratios peuvent indiquer des paramètres TTL incorrects ou des paramètres de requête inutiles

Volume de demandes d'origine

Surveillez les demandes qui atteignent les points de MediaPackage terminaison

Des volumes élevés peuvent indiquer des problèmes de mise en cache

Variations des clés du cache

Passez en revue les modèles de clés du cache pour identifier les variations inutiles

Trop de variations réduisent l'efficacité du cache

Après avoir mis en œuvre ces optimisations du cache, configurez la surveillance pour suivre leur efficacité. Pour obtenir des conseils sur la surveillance des taux de réussite du cache, des modèles de demandes d'origine et d'autres indicateurs de performance clés, consultez [Surveillez les performances pour MediaPackage le CDN et les intégrations MediaTailor](#). Si vous observez des performances médiocres du cache ou des demandes d'origine inattendues, consultez [Résolution des problèmes MediaPackage, CDN et intégrations MediaTailor](#) les étapes de résolution des problèmes.

Surveillez les performances pour MediaPackage le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor nécessite une surveillance efficace pour maintenir des performances optimales de votre intégration AWS Elemental MediaPackage et de celle de votre réseau de diffusion de contenu (CDN). Cette rubrique fournit des conseils sur les indicateurs clés à suivre, les outils de surveillance à utiliser et la façon de configurer des alertes pour une détection proactive des problèmes.

Avant de configurer la surveillance, assurez-vous que votre intégration de base fonctionne correctement. Si vous n'avez pas terminé la configuration de base de l'intégration de votre réseau de diffusion de contenu, commencez par [Intégration MediaTailor avec un MediaPackage CDN](#). Si vous devez résoudre les problèmes identifiés par le biais de la surveillance, consultez.. [Résolution des problèmes d'intégration au CDN](#)

Indicateurs de performance clés

Surveillez ces indicateurs essentiels pour garantir des performances optimales de votre intégration MediaPackage et de celle du CDN :

Indicateurs de performance du CDN

Pour obtenir des indicateurs complets des performances du CDN, notamment les objectifs de taux de réussite du cache, la surveillance du volume des demandes d'origine et les tests de temps de réponse, consultez le [Benchmarks de performance pour le CDN et les intégrations MediaTailor](#) guide d'optimisation du CDN.

Principales considérations spécifiques à l'EMP pour les métriques du CDN :

En-têtes de contrôle du cache EMP

Ce qu'il faut vérifier : assurez-vous que votre CDN respecte les en-têtes de contrôle du cache d'EMP pour un comportement TTL optimal

Comportement attendu : les différents types de contenu doivent avoir des durées de cache différentes en fonction des en-têtes de l'EMP

Pour obtenir des instructions détaillées sur l'optimisation du cache EMP, consultez [Optimisation de la mise en cache du CDN MediaTailor et de la diffusion de contenu MediaPackage](#).

Impact des paramètres de requête

Éléments à surveiller : suivre l'impact des paramètres de requête spécifiques à l'EMP sur l'efficacité du cache

Cible d'optimisation : assurez-vous que seuls les paramètres de requête EMP nécessaires sont inclus dans les clés de cache

Temps de réponse du CDN

Éléments à surveiller : suivez les temps de réponse pour les différents types de contenu (manifestes ou segments).

Valeurs cibles :

- Contenu mis en cache : moins de 100 ms
- Demandes d'origine : moins de 500 ms

MediaPackage mesures de performance

Taux d'erreur

Éléments à surveiller : surveillez les taux d'erreur HTTP à la fois sur votre CDN et sur vos points de MediaPackage terminaison. Portez une attention particulière aux erreurs 4xx, qui peuvent indiquer des problèmes de configuration.

Codes d'erreur clés :

- 400 erreurs : souvent liées à des problèmes de filtrage des manifestes
- Erreurs 404 : cela peut indiquer des problèmes de routage ou de configuration
- Erreurs 504 : problèmes de temporisation, en particulier avec LL-HLS

Volume et modèles de demandes

Éléments à surveiller : suivez les modèles de demandes adressées aux MediaPackage terminaux afin d'identifier les tendances d'utilisation et les besoins en capacité.

Motifs à surveiller :

- Heures de pointe d'utilisation
- Répartition géographique des demandes
- Distribution du type de contenu (en direct ou à la demande)

Métriques de latence

End-to-end latence

Éléments à surveiller : pour les implémentations de LL-HLS, surveillez le temps de end-to-end latence entre l'ingestion du contenu et sa lecture par le lecteur. Une latence élevée peut indiquer des problèmes de configuration du CDN.

Valeurs cibles :

- LL-HLS : moins de 3 secondes de latence glass-to-glass
- HLS normal : moins de 30 secondes

Heure de génération du manifeste

Éléments à surveiller : temps nécessaire MediaPackage à la génération des manifestes, en particulier lorsque le filtrage est appliqué.

Valeurs cibles : moins de 200 ms pour la génération du manifeste

Outils de surveillance et configuration

La mise en place d'outils de surveillance complets est essentielle pour maintenir des performances optimales et identifier rapidement les problèmes avant qu'ils n'affectent les spectateurs. Sans surveillance appropriée, la dégradation des performances, l'inefficacité du cache ou les problèmes d'intégration peuvent passer inaperçus jusqu'à ce que les spectateurs constatent une mauvaise qualité de lecture. La bonne configuration de surveillance offre une visibilité sur tous les aspects de votre intégration MediaPackage et de celle du CDN.

Utilisez ces services et outils AWS pour surveiller votre intégration MediaPackage et celle du CDN :

Amazon CloudWatch

Amazon CloudWatch fournit la base pour surveiller votre intégration MediaPackage et celle du CDN en collectant et en stockant les métriques des deux services. Une CloudWatch configuration appropriée garantit que vous disposez des données nécessaires pour identifier les tendances en matière de performances, résoudre les problèmes et optimiser votre intégration. Sans CloudWatch indicateurs, vous manquerez de visibilité sur les performances du système et vous risquez de ne pas détecter les problèmes avant qu'ils ne deviennent critiques.

Configurez CloudWatch la surveillance pour une collecte complète de métriques :

1. **MediaPackage métriques** : activez CloudWatch les métriques pour vos MediaPackage terminaux afin de suivre les volumes de demandes, les taux d'erreur et les temps de réponse.
2. **Métriques du CDN** : configurez CloudWatch pour collecter des CloudFront métriques, notamment les taux de réussite du cache, le nombre de demandes d'origine et les taux d'erreur.
3. **Mesures personnalisées** : créez des statistiques personnalisées spécifiques à l'entreprise, KPIs telles que l'engagement des spectateurs ou la popularité du contenu.

CloudWatch tableaux de bord

Créez des tableaux de bord complets pour visualiser vos indicateurs :

1. **Tableau de bord général** : indicateurs de haut niveau indiquant l'état général du système
2. **Tableau de bord des performances du CDN** : mesures détaillées du CDN, notamment les performances du cache et la distribution géographique
3. **MediaPackage tableau de bord des performances** : mesures MediaPackage spécifiques, notamment les modèles de demandes et les taux d'erreur

4. Tableau de bord de End-to-end latence : mesures de latence pour différents types de contenu et différentes régions

Analyse du journal

Configurez l'analyse des journaux pour un dépannage détaillé :

1. Journaux d'accès au CDN : activez et analysez les journaux d'accès au CDN pour comprendre les modèles de demandes et identifier les problèmes
2. MediaPackage CloudWatch journaux : surveillez les MediaPackage journaux pour détecter les erreurs et les problèmes de performance
3. Agrégation de journaux : utilisez Amazon CloudWatch Logs Insights ou des outils tiers pour analyser les modèles de journaux

Configurer des alertes et des notifications

La configuration des alertes est essentielle à la détection et à la résolution proactives des problèmes. Sans alertes appropriées, les problèmes peuvent passer inaperçus jusqu'à ce qu'ils aient un impact significatif sur l'expérience des utilisateurs ou qu'ils entraînent des interruptions de service. Des alertes bien configurées vous aident à identifier et à résoudre les problèmes avant qu'ils n'affectent vos spectateurs, et à garantir que votre équipe est informée des problèmes critiques nécessitant une attention immédiate.

Configurez des alertes proactives pour identifier les problèmes avant qu'ils n'affectent les utilisateurs :

Alertes critiques

Configurez des alertes immédiates pour les problèmes critiques :

- Taux d'erreur élevés : alerte lorsque les taux d'erreur 4xx ou 5xx dépassent 5 % sur une période de 5 minutes
- Diminution du taux d'accès au cache : alerte lorsque le taux d'accès au cache tombe en dessous de 70 % pour les manifestes ou de 85 % pour les segments
- Latence élevée : alerte lorsque la end-to-end latence dépasse les seuils cibles
- Pics de demandes d'origine : alerte lorsque les demandes d'origine augmentent de plus de 50 % par rapport à la base de référence

Alertes d'avertissement

Configurez des alertes d'avertissement pour les tendances susceptibles d'indiquer l'apparition de problèmes :

- Dégradation progressive des performances : alerte lorsque les temps de réponse augmentent de 20 % sur une période de 30 minutes
- Tendances en matière d'efficacité du cache : alerte lorsque les taux d'accès au cache indiquent une tendance à la baisse au fil du temps
- Modèles de trafic inhabituels : alerte en cas de modifications inattendues du volume des demandes ou de la distribution géographique

Utiliser les données de surveillance pour l'optimisation

Tirez parti des données de surveillance pour améliorer continuellement les performances :

Évaluations régulières des performances

1. Examens hebdomadaires : analysez les tendances de performance hebdomadaires et identifiez les opportunités d'optimisation
2. Planification mensuelle de la capacité : utilisez les modèles de trafic pour planifier les besoins en capacité et l'optimisation du CDN
3. Révisions trimestrielles de l'architecture : évaluez l'efficacité globale de l'architecture et identifiez les domaines à améliorer

Actions d'optimisation courantes

Sur la base des données de surveillance, considérez les actions d'optimisation suivantes :

- Ajustements de la politique de cache : modifiez les valeurs TTL en fonction des modèles de mise à jour du contenu réels. Pour obtenir des conseils détaillés sur l'optimisation TTL, consultez [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#).
- Optimisation géographique : ajoutez des emplacements périphériques CDN dans les régions à fort trafic
- Optimisation des paramètres de requête : supprimez les paramètres de requête inutiles qui fragmentent le cache

- Configuration du bouclier d'origine : implémentez le bouclier d'origine dans les régions où le volume de demandes d'origine est élevé

Pour des instructions de surveillance détaillées spécifiques à MediaPackage, voir [Surveillance MediaPackage](#) dans le guide de MediaPackage l'utilisateur.

Résolution des problèmes MediaPackage, CDN et intégrations MediaTailor

AWS Elemental MediaTailor l'intégration au réseau de diffusion de contenu (CDN) AWS Elemental MediaPackage et le réseau de diffusion de contenu (CDN) peuvent rencontrer des problèmes courants qui affectent la lecture, la mise en cache ou d'autres fonctionnalités d'intégration. Utilisez ce guide lorsque vous rencontrez des problèmes de lecture, de mise en cache ou d'autres erreurs liées à l'intégration.

Pour obtenir des conseils complets sur le dépannage du CDN, notamment sur les problèmes de performances du cache universel, la résolution des erreurs HTTP, les procédures de test et les techniques de diagnostic applicables à toutes les MediaTailor implémentations, consultez [Résoudre les problèmes liés à l'intégration du CDN](#). Cette section se concentre sur les exigences MediaPackage spécifiques en matière de dépannage.

Avant de résoudre le problème, assurez-vous d'avoir correctement effectué la configuration de base de l'intégration. Si vous n'avez pas encore configuré votre intégration ou si vous devez passer en revue les étapes de configuration, consultez [Intégration MediaTailor avec un MediaPackage CDN](#). Pour obtenir des conseils sur l'optimisation des performances du cache une fois les problèmes résolus, consultez [Mise en cache du CDN](#).

Erreurs de filtrage manifestes

Problèmes liés à la fonctionnalité de filtrage des MediaPackage manifestes, sur la base de conditions d'erreur documentées :

Erreurs HTTP 400 avec filtrage des manifestes

Symptômes : les requêtes avec `aws.manifestfilter` paramètres renvoient une mauvaise demande HTTP 400

Causes validées (tirées de AWS la documentation) :

- Les critères de filtrage donnent lieu à un manifeste vide (aucun flux ne correspond aux conditions du filtre)

- Noms ou valeurs de paramètres de filtre non valides
- Syntaxe de chaîne de requête mal formée
- Paramètres de filtre dupliqués ou répétés
- La chaîne de paramètres du filtre dépasse 1 024 caractères
- Paramètres de requête appliqués aux playlists ou aux segments multimédias (non pris en charge)

Solutions :

1. Vérifiez les paramètres de vos filtres pour vous assurer qu'ils correspondent aux flux de contenu disponibles. Si le filtrage n'aboutit à aucun flux correspondant, MediaPackage renvoie HTTP 400.
2. Validez la syntaxe du filtre par rapport aux noms de paramètres et aux formats de valeurs pris en charge.
3. Vérifiez qu'il n'y a pas de doublons dans votre chaîne de requête.
4. Assurez-vous que les paramètres du filtre ne sont appliqués qu'aux listes de lecture multivariantes, et non aux playlists ou aux segments multimédias.
5. Vérifiez que votre chaîne de paramètres totale est inférieure à 1 024 caractères.

Référence : [conditions d'erreur de filtrage du AWS Elemental MediaPackage manifeste](#)

Le filtrage des manifestes ne fonctionne pas (HTTP 200 mais aucun filtrage n'est appliqué)

Symptômes : les requêtes renvoient HTTP 200 mais le manifeste contient tous les flux au lieu d'un sous-ensemble filtré

Causes possibles :

- Le CDN ne transmet pas le paramètre `aws.manifestfilter` de requête à MediaPackage
- Paramètre de filtre introuvable dans les flux disponibles (renvoie un manifeste non filtré avec HTTP 200)

Solutions :

1. Vérifiez que votre politique de cache CDN figure `aws.manifestfilter` dans la liste des chaînes de requête transférées.
2. Testez les paramètres du filtre directement par rapport aux MediaPackage points de terminaison (en contournant le CDN) pour vérifier qu'ils fonctionnent comme prévu.
3. Vérifiez que les valeurs des filtres correspondent aux caractéristiques réelles de vos flux de contenu.

Référence : [conditions d'erreur de filtrage du AWS Elemental MediaPackage manifeste](#)

Procédures de diagnostic

Les procédures de diagnostic systématiques vous aident à identifier rapidement et efficacement la cause première des problèmes d'intégration. L'adoption d'une approche structurée évite de perdre du temps sur des hypothèses erronées et garantit que vous abordez le problème réel plutôt que les symptômes. Ces étapes de diagnostic fondées sur des preuves sont conçues pour isoler les problèmes et vous guider vers la solution appropriée.

Suivez ces étapes diagnostiques fondées sur des données probantes pour identifier les problèmes :

Analyser les performances du cache

L'analyse des performances du cache est cruciale pour les intégrations EMP, car une faible efficacité du cache entraîne une augmentation de la charge d'origine, des coûts plus élevés et des problèmes de lecture potentiels.

Pour un dépannage complet des performances du cache, notamment une analyse du taux de réussite du cache, une optimisation des clés de cache et des étapes de diagnostic systématiques, consultez [Problèmes de performance du cache CDN](#) le guide de dépannage principal du CDN.

Considérations relatives au cache spécifiques à l'EMP :

- En-têtes de contrôle de cache EMP : vérifiez que votre CDN respecte les en-têtes de contrôle de cache d'EMP plutôt que de les remplacer
- Paramètres de requête EMP : assurez-vous que seuls les paramètres de requête EMP nécessaires sont inclus dans les clés de cache
- Comportement TTL EMP : vérifiez que les différents types de contenu EMP ont des durées de cache appropriées

Pour obtenir des instructions détaillées sur l'optimisation des politiques de cache EMP et des paramètres TTL, consultez. [Optimisation de la mise en cache du CDN MediaTailor et de la diffusion de contenu MediaPackage](#)

Valider la configuration du filtrage du manifeste

La validation du filtrage des manifestes est essentielle, car les problèmes de filtrage peuvent entraîner la réception de contenus incorrects, de formats non pris en charge ou de contenus auxquels

ils ne devraient pas avoir accès. Les tests systématiques permettent de déterminer si les problèmes sont liés à la configuration du CDN, à la syntaxe des paramètres de filtre ou à la disponibilité du contenu.

Testez systématiquement la fonctionnalité de filtrage des manifestes :

1. Testez les paramètres du filtre directement par rapport aux MediaPackage points de terminaison (en contournant le CDN) pour vérifier qu'ils fonctionnent correctement.
2. Comparez les manifestes filtrés et non filtrés pour confirmer que les flux attendus sont inclus/exclus.
3. Vérifiez que votre politique de cache CDN transmet le paramètre de `aws.manifestfilter` requête.
4. Vérifiez les erreurs HTTP 400 et comparez-les aux conditions d'erreur documentées.

Si vous devez implémenter ou modifier le filtrage des manifestes après avoir résolu les problèmes, consultez [Configurer le filtrage des manifestes avec MediaTailor MediaPackage, et le CDN](#) les instructions de configuration complètes.

Valider la configuration des paramètres de requête

Assurez-vous que votre CDN transmet uniquement les paramètres de requête requis :

1. Passez en revue votre politique de cache CDN pour vous assurer qu'elle inclut uniquement les paramètres AWS recommandés :
 - `aws.manifestfilter`- pour le filtrage des manifestes
 - `aws.manifestsettings`- pour une visualisation décalée dans le temps
 - `_HLS_msnet_HLS_part` - pour le support LL-HLS
2. Supprimez tous les autres paramètres de requête de votre clé de cache, car ils sont MediaPackage ignorés et réduisent l'efficacité du cache.

Référence : [Travailler avec AWS Elemental MediaPackage et CDNs](#)

Référence du code d'erreur

Référence pour les conditions d'erreur documentées et leurs causes :

Mauvaise requête HTTP 400 (filtrage des manifestes)

Causes documentées :

- L'application du filtre entraîne un manifeste vide
- Noms ou valeurs de paramètres non valides
- Syntaxe de chaîne de requête mal formée
- Paramètres de filtre dupliqués
- La chaîne de paramètres dépasse 1 024 caractères
- Paramètres de requête sur des playlists ou des segments multimédias

Référence : [conditions d'erreur de filtrage du MediaPackage manifeste](#)

HTTP 200 OK (aucun filtrage appliqué)

Causes documentées :

- Paramètre de filtre introuvable dans les flux disponibles (renvoie un manifeste non filtré)
- Seuls les flux de sous-titres sont présents après le filtrage (renvoie un manifeste non filtré)

Référence : [conditions d'erreur de filtrage du MediaPackage manifeste](#)

Ressources supplémentaires pour la résolution des problèmes

Pour les questions non abordées dans cette rubrique, consultez les AWS ressources officielles suivantes :

- [Prévisualisation d'un manifeste à partir de AWS Elemental MediaPackage](#) - Utiliser l'aperçu du manifeste pour résoudre les problèmes d'emballage de contenu
- [Augmenter CloudFront le taux de réussite du cache](#) - Guide complet sur l'optimisation du cache du CDN
- [Filtrage des manifestes](#) - Guide complet des fonctionnalités MediaPackage de filtrage

Intégration AWS Elemental MediaTailor à Amazon CloudFront

AWS Elemental MediaTailor s'intègre à Amazon CloudFront pour améliorer les performances et la fiabilité de diffusion de contenu. CloudFront est un réseau de diffusion de contenu (CDN) qui distribue votre contenu par le biais d'un réseau mondial de centres de données appelés emplacements

périphériques. Lorsque les internautes demandent votre contenu MediaTailor, CloudFront achemine les demandes vers l'emplacement périphérique le plus proche. Cette approche réduit le temps de latence et améliore les performances de vos spectateurs.

L'intégration MediaTailor avec CloudFront offre plusieurs avantages clés :

- Temps de latence réduit pour les spectateurs qui accèdent à du contenu personnalisé
- Évolutivité améliorée pour gérer un public de grande taille
- Fiabilité accrue grâce à des voies de livraison redondantes
- Optimisation des coûts grâce à des stratégies de mise en cache efficaces
- Des fonctionnalités avancées telles que le basculement multirégional avec Media Quality-Aware Resiliency (MQAR)

Pour obtenir des informations complètes sur CloudFront les fonctionnalités, consultez le [guide du CloudFront développeur](#). Pour plus d'informations sur la CloudFront tarification, consultez la section [CloudFront Tarification](#).

Par souci de concision, nous utilisons parfois des « manifestes » pour désigner collectivement des listes de lecture multivariantes, des listes de lecture multimédia et. MPDs

Rubriques

- [Configurez CloudFront l'intégration de base avec MediaTailor](#)
- [Optimisation MediaTailor des performances grâce aux CloudFront fonctionnalités](#)
- [Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR](#)
- [Surveillez et dépannez votre CloudFront intégration MediaTailor](#)

Configurez CloudFront l'intégration de base avec MediaTailor

AWS Elemental MediaTailor l'intégration avec Amazon CloudFront améliore les performances de diffusion de contenu pour vos spectateurs. Cette rubrique vous explique comment configurer une CloudFront distribution de base pour MediaTailor. Grâce à cette intégration, vos spectateurs peuvent accéder à du contenu personnalisé via le CloudFront réseau. Vous apprendrez également à configurer une mise en cache appropriée pour différents types de contenu.

Pour plus d'informations sur la transmission de paramètres de requête à CloudFront des fins d'autorisation et de routage, consultez [MediaTailor paramètres de requête manifeste](#). Pour un routage

avancé à l'aide de variables dynamiques, voir [MediaTailor variables de domaine pour plusieurs sources de contenu](#).

Conditions préalables

Avant de procéder CloudFront à la configuration avec MediaTailor, assurez-vous de disposer des éléments suivants :

- Un AWS compte actif autorisé à créer et à gérer CloudFront des distributions
- Une configuration MediaTailor de lecture configurée (voir [Utilisation AWS Elemental MediaTailor pour insérer des publicités](#))
- Votre serveur d'origine de contenu correctement configuré et accessible
- Compréhension de base des concepts de streaming vidéo (HLS/DASH)

Configuration de CloudFront la distribution

Procédez comme suit pour créer et configurer une CloudFront distribution pour MediaTailor :

Pour créer une CloudFront distribution pour MediaTailor

1. Connectez-vous à la console AWS de gestion et ouvrez-la sur CloudFront <https://console.aws.amazon.com/cloudfront/v3/home>.
2. Choisissez Create Distribution.
3. Pour le domaine Origin, entrez l'URL de votre MediaTailor point de terminaison (par exemple, `a1b2c3d4.mediatailor.us-west-2.amazonaws.com`).
4. Pour Protocole, sélectionnez HTTPS uniquement.
5. Dans Nom, entrez un nom qui vous aide à identifier cette origine (par exemple, `mediataylor-origin`).
6. Configurez les paramètres de comportement du cache par défaut :
 - a. Pour le modèle de chemin, utilisez la valeur par défaut (*).
 - b. Pour compresser automatiquement les objets, sélectionnez Oui.
 - c. Pour la politique du protocole Viewer, sélectionnez Rediriger le HTTP vers HTTPS.
 - d. Pour les méthodes HTTP autorisées, sélectionnez GET, HEAD.
 - e. Pour la politique de cache, sélectionnez CachingDisabled.

- f. Pour la politique de demande Origin, sélectionnez AllViewer cette option pour transférer tous les en-têtes pour le comportement par défaut.

 Note

Le comportement par défaut permet AllViewer de gérer en toute sécurité tout contenu qui ne correspond pas à des modèles de chemin spécifiques. Les comportements de cache spécifiques pour les manifestes et les segments seront configurés séparément avec des politiques appropriées.

7. Configurez les paramètres de distribution :
 - a. Pour la catégorie de prix, sélectionnez l'option qui correspond le mieux à la zone géographique de votre public.
 - b. Pour l'ACL AWS WAF Web, sélectionnez une ACL Web existante ou laissez le champ Ne pas activer les protections de sécurité.
 - c. Pour l'objet racine par défaut, laissez le champ vide.
 - d. Pour la journalisation standard, sélectionnez Activé pour activer la journalisation.
8. Choisissez Create Distribution.

Configuration des comportements du cache

Après avoir créé votre distribution, vous devez configurer des comportements de cache supplémentaires pour gérer les différents types de contenu de manière appropriée. Cette section décrit la configuration de base du comportement du cache pour CloudFront.

Pour une optimisation complète de la mise en cache, y compris les paramètres TTL avancés, les configurations des clés de cache et le réglage des performances, consultez [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#) le guide d'optimisation du CDN.

Configuration du comportement du cache du manifeste

Ne mettez pas en cache les manifestes car ils contiennent du contenu personnalisé. Pour configurer le comportement du cache, procédez comme suit :

Pour configurer le comportement du cache du manifeste

1. Dans la CloudFront console, sélectionnez votre distribution.

2. Choisissez l'onglet Comportements.
3. Choisissez Create behavior (Créer un comportement).
4. Dans le champ Path pattern, saisissez pour `*.m3u8` qu'il corresponde à des playlists HLS multivariantes et à des playlists multimédias.
5. Pour Origin, sélectionnez votre MediaTailor origine.
6. Pour la politique de cache, sélectionnez CachingDisabled.
7. Pour la politique de demande Origin, sélectionnez AllViewercette option pour transférer tous les en-têtes requis pour le contenu dynamique.
8. Choisissez Créer.
9. Répétez ces étapes pour les manifestes DASH en utilisant le modèle `*.mpd` de chemin correspondant MPDs.

Cette configuration garantit que chaque spectateur reçoit un manifeste personnalisé avec son contenu publicitaire spécifique. Le CDN ne met pas ces manifestes en cache. Chaque demande est donc directement envoyée à MediaTailor.

Configuration du comportement du cache de segments

Configurez des comportements de cache distincts pour les segments publicitaires et les segments de contenu afin d'optimiser les performances et de garantir une gestion appropriée du CORS.

Configuration du comportement du cache des segments publicitaires

Les segments publicitaires diffusés selon le modèle de `/tm/*` chemin nécessitent une configuration spécifique pour gérer correctement le CORS. Procédez comme suit :

Pour configurer le comportement du cache des segments publicitaires

1. Dans la CloudFront console, sélectionnez votre distribution.
2. Choisissez l'onglet Comportements.
3. Choisissez Create behavior (Créer un comportement).
4. Dans le champ Path pattern, saisissez `/tm/*` pour qu'il corresponde aux segments publicitaires diffusés par MediaTailor.
5. Pour Origin, sélectionnez l'origine de vos MediaTailor segments (en utilisant le `segments.mediatailor.region.amazonaws.com` nom d'hôte).
6. Pour la politique de cache, sélectionnez CachingOptimized.

7. Pour la politique de demande d'origine, sélectionnez Aucune.
8. Pour la politique des en-têtes de réponse, sélectionnez CORS- with-preflight-and - SecurityHeadersPolicy pour vous assurer que les en-têtes CORS appropriés sont inclus dans les réponses.
9. Choisissez Créer.

Configuration du comportement du cache des segments de contenu

Les segments de contenu peuvent utiliser des politiques de mise en cache standard pour des performances optimales. Configurez des comportements distincts pour les différents formats de segment :

Pour configurer le comportement du cache des segments de contenu

1. Dans la CloudFront console, sélectionnez votre distribution.
2. Choisissez l'onglet Comportements.
3. Choisissez Create behavior (Créer un comportement).
4. Pour le modèle de chemin, entrez `*.ts` pour correspondre aux segments de contenu HLS.
5. Pour Origin, sélectionnez l'origine de votre contenu.
6. Pour la politique de cache, sélectionnez CachingOptimized.
7. Pour la politique de demande d'origine, sélectionnez Aucune.
8. Pour la politique des en-têtes de réponse, sélectionnez CORS- with-preflight-and - SecurityHeadersPolicy pour garantir une gestion CORS cohérente de tous les types de contenu.
9. Choisissez Créer.
10. Répétez ces étapes pour les autres formats de segments de contenu en utilisant les modèles de chemin appropriés :
 - `*.mp4` pour les MP4 segments
 - `*.m4s` pour les segments DASH
 - `*.cmfv` et `*.cmfa` pour les segments CMAF

Cette configuration garantit que les segments publicitaires et les segments de contenu sont mis en cache de manière appropriée avec une gestion CORS appropriée. Les segments publicitaires utilisent l'origine MediaTailor des segments avec une protection CORS, tandis que les segments de contenu utilisent votre origine de contenu avec des politiques de mise en cache optimisées.

Mettre à jour MediaTailor la configuration

Après avoir configuré votre CloudFront distribution, mettez à jour votre MediaTailor configuration pour utiliser le CloudFront domaine :

Pour mettre à jour votre MediaTailor configuration

1. Ouvrez la [MediaTailor console](#).
2. Sélectionnez la configuration que vous souhaitez mettre à jour.
3. Dans la section Configuration du CDN, entrez le nom de votre domaine de CloudFront distribution (par exemple, `d1234abcdef.cloudfront.net`) dans le champ de préfixe du segment de contenu CDN.
4. Enregistrez vos modifications.

Avec cette configuration, MediaTailor génère des manifestes pointant vers votre CloudFront distribution plutôt URLs que directement vers l'origine.

Tester votre intégration

Après avoir configuré votre CloudFront distribution et mis à jour votre MediaTailor configuration, testez l'intégration :

Pour tester votre MediaTailor intégration CloudFront et votre intégration

1. Demandez un manifeste par le biais CloudFront de votre distribution (par exemple, `https://d1234abcdef.cloudfront.net/v1/master/12345/my-config/index.m3u8`).
2. Vérifiez que le manifeste contient un segment URLs qui pointe vers votre CloudFront domaine.
3. Diffusez le contenu via un lecteur vidéo et vérifiez que le contenu et les publicités s'affichent correctement.
4. Vérifiez les CloudFront journaux pour vous assurer que les demandes sont correctement acheminées.

Exemple de configuration

Voici un exemple de configuration de CloudFront distribution MediaTailor avec des comportements de cache appropriés :

Exemple CloudFront exemple de configuration de distribution

```
{
  "DefaultCacheBehavior": {
    "TargetOriginId": "mediatailor-origin",
    "ViewerProtocolPolicy": "redirect-to-https",
    "AllowedMethods": {
      "Quantity": 2,
      "Items": ["GET", "HEAD"]
    },
    "CachePolicyId": "4135ea2d-6df8-44a3-9df3-4b5a84be39ad",
    "OriginRequestPolicyId": "59781a5b-3903-41f3-afcb-af62929ccde1",
    "Comment": "Default behavior with CachingDisabled and AllViewer"
  },
  "CacheBehaviors": [
    {
      "PathPattern": "/*.m3u8",
      "TargetOriginId": "mediatailor-origin",
      "ViewerProtocolPolicy": "redirect-to-https",
      "CachePolicyId": "4135ea2d-6df8-44a3-9df3-4b5a84be39ad",
      "OriginRequestPolicyId": "59781a5b-3903-41f3-afcb-af62929ccde1",
      "Comment": "Manifest behavior with CachingDisabled and AllViewer"
    },
    {
      "PathPattern": "/*.ts",
      "TargetOriginId": "mediatailor-origin",
      "ViewerProtocolPolicy": "redirect-to-https",
      "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e58f6",
      "OriginRequestPolicyId": "88a5eaf4-2fd4-4709-b370-b4c650ea3fcf",
      "Comment": "Segment behavior with CachingOptimized and HostHeaderOnly"
    }
  ]
}
```

Cet exemple montre :

- Comportement par défaut : utilise CachingDisabled et AllViewer gère en toute sécurité tout contenu qui ne correspond pas à des modèles de chemin spécifiques
- Comportement du manifeste (*.m3u8) : utilisations CachingDisabled et pour le contenu dynamique AllViewer

- Comportement du segment (*.ts) : utilisations `CachingOptimized` et `CORS-with-preflight-and-SecurityHeadersPolicy`

Étapes suivantes

Après avoir configuré votre CloudFront intégration de base avec MediaTailor, considérez les étapes suivantes :

- Optimisez les performances grâce à des CloudFront fonctionnalités supplémentaires (voir [Optimisation MediaTailor des performances grâce aux CloudFront fonctionnalités](#))
- Mettre en œuvre la résilience multirégionale avec MQAR (voir [Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR](#))
- Configurer la surveillance et le dépannage (voir [Surveillez et dépannez votre CloudFront intégration MediaTailor](#))

Optimisation MediaTailor des performances grâce aux CloudFront fonctionnalités

AWS Elemental MediaTailor les performances avec Amazon CloudFront peuvent être améliorées grâce à des fonctionnalités supplémentaires allant au-delà de la configuration de base. Après avoir configuré votre CloudFront configuration de base avec MediaTailor, vous pouvez implémenter des fonctionnalités supplémentaires pour améliorer les performances, la fiabilité et les options de personnalisation. Ces optimisations contribuent à améliorer l'expérience visuelle. Ils réduisent également les coûts et la charge d'origine.

Pour l'optimisation des performances à l'aide du routage dynamique et de la gestion des paramètres, voir [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour les stratégies d'optimisation des paramètres de requête, voir [MediaTailor paramètres de requête manifeste](#).

Réduction de la charge d'origine avec Origin Shield

Origin Shield ajoute une couche de mise en cache entre les emplacements CloudFront périphériques et votre serveur d'origine. Cette fonctionnalité est utile pour la diffusion en direct et les contenus VOD populaires. Cela aide lorsque de nombreux spectateurs demandent le même contenu en même temps.

En consolidant les demandes provenant de plusieurs emplacements périphériques, Origin Shield réduit la charge de travail MediaTailor et l'origine de votre contenu.

Pour activer Origin Shield pour votre MediaTailor origine

1. Ouvrez la CloudFront console et accédez à votre distribution.
2. Sélectionnez l'origine qui pointe vers votre configuration MediaTailor de lecture.
3. Sous Origin Shield, sélectionnez Oui.
4. Dans le menu déroulant, sélectionnez la AWS région la plus proche de votre MediaTailor terminal.
5. Enregistrez vos modifications.

En cas d'événements à fort trafic, Origin Shield réduit considérablement la charge de requêtes sur votre serveur d'origine et améliore la fiabilité. Pour obtenir des instructions détaillées, consultez la section [Enabling Origin Shield](#) dans le guide du CloudFront développeur.

Personnalisation de la diffusion de contenu avec Functions CloudFront

CloudFront Les fonctions vous permettent d'exécuter JavaScript du code léger à la périphérie pour modifier les demandes et les réponses des utilisateurs. Vous pouvez utiliser ces fonctions pour des personnalisations simples telles que les modifications d'URL, la manipulation d'en-têtes ou l'authentification de base. Pour les MediaTailor flux de travail, les fonctions facilitent les tâches qui ne nécessitent pas de traitement complexe.

CloudFront Les fonctions offrent un moyen léger de personnaliser la diffusion de contenu à la périphérie. Voici comment les implémenter dans le cadre de votre MediaTailor intégration :

Pour implémenter CloudFront des fonctions pour MediaTailor

1. Dans la CloudFront console, accédez à Functions.
2. Créez une nouvelle fonction et sélectionnez l'objectif approprié :
 - Manipulation d'URL : pour modifier une liste de lecture multivariante, une liste de lecture multimédia et une demande MPD URLs avant qu'elles n'atteignent MediaTailor
 - Manipulation des en-têtes : pour ajouter ou modifier des en-têtes de demande
 - Authentification simple : pour valider les jetons ou les paramètres de requête
3. Rédigez le code JavaScript de votre fonction.
4. Testez votre fonction à l'aide d'exemples de MediaTailor requêtes.
5. Publiez et associez la fonction au comportement du cache de votre distribution.

Exemple Exemple de CloudFront fonction pour la normalisation des URL

```
function handler(event) {
    var request = event.request;
    var uri = request.uri;

    // Normalize URLs to lowercase to improve cache hit ratio
    if (uri.includes('.m3u8') || uri.includes('.mpd')) {
        request.uri = uri.toLowerCase();
    }

    return request;
}
```

Pour plus d'informations et des exemples de code, voir [Personnaliser en périphérie à l'aide CloudFront des fonctions](#) dans le guide du CloudFront développeur.

Implémentation de personnalisations avancées avec Lambda @Edge

Lorsque vous avez besoin de capacités de traitement plus complexes que celles CloudFront proposées par Functions, utilisez Lambda @Edge. Ce service vous permet d'exécuter des fonctions Node.js ou Python à des emplacements CloudFront périphériques.

Les fonctions Lambda @Edge peuvent effectuer des opérations sophistiquées telles qu'une authentification complexe, des modifications de réponse plus importantes ou des intégrations d'API tierces.

Pour des personnalisations plus complexes, utilisez les fonctions Lambda @Edge avec MediaTailor votre CloudFront intégration et :

Pour implémenter Lambda @Edge avec MediaTailor

1. Créez une fonction Lambda dans la région USA Est (Virginie du Nord).
2. Rédigez votre code de fonction pour l'un des cas d'utilisation suivants :
 - Manipulation d'URL : pour modifier une liste de lecture multivariante, une liste de lecture multimédia et une demande MPD URLs avant qu'elles n'atteignent MediaTailor
 - Tests A/B - Pour rediriger les utilisateurs vers différents serveurs de décision publicitaire

- Demande d'authentification : pour ajouter des en-têtes d'authentification
 - Modification de l'en-tête de réponse - Pour ajouter des en-têtes CORS
3. Publiez une version de votre fonction et créez un alias de fonction.
 4. Associez la fonction à votre CloudFront distribution au point de déclenchement approprié (demande ou réponse du spectateur).

Note

Lorsque vous utilisez Lambda @Edge avec MediaTailor, évitez d'utiliser des déclencheurs orientés vers l'origine (demande d'origine et réponse d'origine) si vous prévoyez d'utiliser les fonctionnalités Media Quality-Aware Resiliency (MQAR), car elles sont actuellement incompatibles.

Pour plus d'informations et des exemples de code, voir [Personnaliser en périphérie à l'aide de Lambda @Edge](#) dans le guide du CloudFront développeur.

Conseils supplémentaires pour l'optimisation des performances

Envisagez ces optimisations supplémentaires pour améliorer encore les performances :

Optimisez les taux de réussite du cache

Surveillez le taux de réussite de votre cache à l'aide de CloudFront métriques et recherchez des opportunités pour l'améliorer :

- Standardisez les modèles d'URL pour améliorer la cohérence des clés de cache
- Utilisez la liste blanche des chaînes de requête pour inclure uniquement les paramètres nécessaires dans la clé de cache
- Envisagez d'implémenter la normalisation des URL avec CloudFront Functions

Réduisez la latence

Mettez en œuvre les techniques suivantes pour minimiser le temps de latence :

- Activer la compression Brotli pour les réponses basées sur du texte
- Utilisez HTTP/2 ou HTTP/3 pour améliorer l'efficacité de la connexion
- Envisagez d'activer IPv6 le support pour les réseaux modernes

Optimisation des coûts

Trouvez le juste équilibre entre performance et rentabilité :

- Utilisez Origin Shield pour réduire les demandes d'origine redondantes
- Envisagez de sélectionner une classe de prix en fonction de la géographie de votre public
- Mettez en œuvre une mise en cache agressive pour les segments afin de réduire le trafic d'origine

Étapes suivantes

Après avoir optimisé les performances grâce aux CloudFront fonctionnalités, considérez les étapes suivantes :

- Mettre en œuvre la résilience multirégionale avec MQAR (voir [Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR](#))
- Configurer la surveillance et le dépannage (voir [Surveillez et dépannez votre CloudFront intégration MediaTailor](#))

Mettre en œuvre la résilience multirégionale MediaTailor avec MQAR

AWS Elemental MediaTailor La résilience multirégionale est renforcée grâce à Media Quality-Aware Resiliency (MQAR), une CloudFront fonctionnalité avancée d'Amazon qui permet d'offrir la meilleure expérience de streaming possible. Il sélectionne automatiquement l'origine ayant le score de qualité le plus élevé lorsque vous avez plusieurs origines dans différentes AWS régions. Cette fonctionnalité est particulièrement utile pour la diffusion en direct où vous avez besoin d'un service ininterrompu.

Comment fonctionne MQAR

MQAR permet CloudFront de sélectionner automatiquement l'origine ayant le score de qualité le plus élevé. Cela garantit aux spectateurs la meilleure expérience de streaming possible. Lorsqu'il est configuré correctement, le MQAR offre les avantages suivants :

- Sélection automatique de l'origine de la plus haute qualité
- Basculement fluide entre les AWS régions en cas de panne
- Expérience utilisateur améliorée avec un minimum d'interruptions
- Real-time suivi de la qualité et adaptation

Étape 1 : vérifier les exigences en matière de MQAR

Avant de mettre en œuvre le MQAR, vérifiez que votre infrastructure répond à ces exigences. Le MQAR fonctionne en comparant les scores de qualité de plusieurs origines. Vous avez donc besoin d'un contenu identique disponible dans plusieurs AWS régions.

- Encodeurs envoyant des flux d'ingestion alignés à tous les MediaPackage canaux à l'aide de flux d'ingestion CMAF verrouillés par époque
- Deux MediaPackage canaux identiques dans différentes AWS régions, avec des points de terminaison d'origine identiques
- Ingestion CMAF pour les MediaPackage canaux (activée par défaut)
- CloudFront distribution configurée pour prendre en charge MQAR
- MediaTailor configurations pour chaque MediaPackage terminal

Étape 2 : Configurez vos encodeurs pour MQAR

Vos encodeurs doivent produire des sorties cohérentes et synchronisées dans toutes les régions pour que le MQAR fonctionne efficacement. Cette cohérence permet de CloudFront faire des comparaisons de qualité précises entre les origines.

Configurez vos MediaLive sorties avec les paramètres suivants :

- Assurez-vous que toutes les fréquences d'images vidéo du groupe de sortie CMAF sont cohérentes (toutes les fréquences d'images fractionnaires ou entières).
- Évitez les transitions entre les fréquences d'images fractionnaires et entières au cours des sessions de codage.
- Configurez les numéros de séquence des segments de sortie afin qu'ils ne reviennent jamais en arrière d'une session de codage à l'autre.
- Utilisez des noms de sortie d'encodeur identiques dans toutes les régions.

Pour plus d'informations sur la configuration du MediaLive MQAR, consultez la section [Utilisation du MQCS](#) dans le guide de l' MediaLive utilisateur.

Étape 3 : Configuration MediaPackage pour MQAR

Configurez vos MediaPackage chaînes et vos terminaux à l'aide des configurations suivantes :

Pour configurer MediaPackage pour MQAR

1. Créez des configurations de canaux et de terminaux identiques dans chaque AWS région.
2. Utilisez CMAF comme type d'entrée de canal.
3. Pour l' MediaPackage origine principale, activez la configuration des erreurs de point de terminaison forcée avec les paramètres suivants :
 - Listes de lecture multivariantes, listes de lecture multimédia ou MPD périmés
 - Liste de lecture multivariante, playlist multimédia ou MPD incomplète
 - Entrée Slate
4. Pour les MediaPackage origines de sauvegarde, n'activez pas ces configurations d'erreur afin de maximiser les chances de réussite du basculement.

Pour plus d'informations sur la configuration MediaPackage pour MQAR, consultez la section [Tirer parti des scores de qualité multimédia AWS Elemental MediaPackage](#) à l'aide du guide de l' MediaPackage utilisateur.

Étape 4 : Configuration CloudFront pour MQAR

Dans la CloudFront configuration, vous activez MQAR et définissez comment il sélectionne vos origines. Créez un groupe d'origine avec l'option Media quality score activée.

Pour configurer CloudFront pour MQAR

1. Créez une CloudFront distribution dont les origines pointent vers vos MediaTailor points de terminaison.
2. Créez un groupe d'origine qui inclut ces origines.
3. Dans les paramètres du groupe d'origine, activez l'option Score de qualité multimédia.
4. Configurez les critères de basculement pour inclure les codes de réponse 404 Not Found. Vous pouvez éventuellement inclure d'autres 4xx/5xx codes.
5. Créez des comportements de cache distincts pour le modèle de chemin de chaque canal. Cela empêche le mixage des partitions lorsque le même groupe d'origine dessert plusieurs canaux.

 Note

Le MQAR n'est pas disponible lorsque vous utilisez les fonctions Lambda @Edge dans des déclencheurs orientés vers l'origine (demande d'origine et réponse d'origine) associés au comportement du cache de votre distribution.

Pour plus d'informations sur la configuration CloudFront pour MQAR, consultez la section Résilience tenant compte de la qualité des [médias](#) dans le guide du développeur. CloudFront

Étape 5 : Configuration MediaTailor pour MQAR

Pour terminer la configuration de votre MQAR, configurez chaque MediaTailor région pour qu'elle fonctionne avec votre architecture multirégionale. Cela garantit une insertion d'annonces cohérente, quelle que soit l'origine CloudFront sélectionnée.

Pour configurer MediaTailor pour MQAR

1. Créez des MediaTailor configurations identiques dans chaque région, en pointant vers les MediaPackage extrémités correspondantes.
2. Configurez le préfixe du segment de contenu CDN pour utiliser votre domaine de CloudFront distribution.
3. Assurez-vous que les configurations du serveur de décision publicitaire sont identiques d'une région à l'autre.

Cette configuration garantit que, quelle que soit l'origine CloudFront sélectionnée en fonction des scores de qualité, MediaTailor vous pouvez continuer à personnaliser les publicités de manière cohérente.

Étape 6 : Testez votre configuration MQAR

Après avoir configuré MQAR, testez votre configuration pour vous assurer qu'elle fonctionne comme prévu :

Pour tester votre configuration MQAR

1. Demandez du contenu par le biais CloudFront de votre distribution.
2. Surveillez les journaux en temps réel pour vérifier CloudFront que les origines sont sélectionnées en fonction des scores de qualité.

3. Simulez une panne dans votre AWS région principale pour tester le comportement de basculement.
4. Vérifiez que l'insertion d'annonces continue de fonctionner correctement pendant le basculement.

Étapes suivantes

Après avoir implémenté le MQAR, considérez les étapes suivantes :

- Configurez la surveillance et le dépannage pour votre configuration MQAR (voir [Surveillez et dépannez votre CloudFront intégration MediaTailor](#))
- Mettez en œuvre un déploiement automatique à l'aide de AWS CloudFormation (voir [Automatisez MediaTailor et créez un CDN avec CloudFormation](#))

Surveillez et dépannez votre CloudFront intégration MediaTailor

AWS Elemental MediaTailor l'intégration avec Amazon CloudFront nécessite une surveillance et un dépannage continus pour maintenir des performances optimales. Après avoir mis en œuvre votre MediaTailor intégration CloudFront et, surveillez les performances et préparez-vous à résoudre les problèmes éventuels. Le réseau de diffusion de contenu (CDN) fournit des outils qui vous aident à comprendre les performances de votre distribution et à identifier les problèmes potentiels.

Configuration de la surveillance pour votre intégration

Mettez en œuvre ces stratégies de surveillance pour suivre les performances de votre CloudFront MediaTailor intégration réseau :

Pour configurer la surveillance de votre MediaTailor intégration CloudFront et

1. Activez CloudFront la journalisation standard :
 - a. Dans la CloudFront console, sélectionnez votre distribution.
 - b. Sélectionnez l'onglet Logs (Journaux).
 - c. Sous Journaux standard, choisissez Modifier.
 - d. Sélectionnez Activé et configurez un compartiment Amazon S3 pour le stockage des journaux.
 - e. Sélectionnez Enregistrer les modifications.

2. Configurez des journaux CloudFront en temps réel :

- a. Dans la CloudFront console, sélectionnez votre distribution.
- b. Sélectionnez l'onglet Logs (Journaux).
- c. Sous Real-time Journaux, choisissez Modifier.
- d. Sélectionnez Activé et configurez un flux de diffusion Amazon Kinesis Data Streams ou Amazon Data Firehose.
- e. Incluez les champs suivants dans la configuration de votre journal :
 - `time-to-first-byte`- Temps de réponse
 - `sc-status`- Code d'état HTTP
 - `c-ip`- Adresse IP du visualiseur
 - `cs-uri-stem`- Demander le chemin de l'URI
 - `cs-user-agent`- Agent utilisateur
 - `x-edge-result-type`- Type de résultat (succès, échec, erreur)
- f. Sélectionnez Enregistrer les modifications.

3. Créez des CloudWatch tableaux de bord :

- a. Dans la CloudWatch console, choisissez Tableaux de bord.
- b. Choisissez Create dashboard (Créer un tableau de bord).
- c. Ajoutez des widgets pour les CloudFront statistiques suivantes :
 - Requêtes
 - BytesDownloaded
 - 4xx ErrorRate
 - 5xx ErrorRate
 - TotalErrorRate
 - CacheHitRate

4. Configurez CloudWatch les alarmes :

- a. Dans la CloudWatch console, choisissez Alarmes.
- b. Sélectionnez Créer une alerte.
- c. Créez des alarmes pour les conditions suivantes :

- 5xx ErrorRate > 1 % pendant 5 minutes
- 4xx ErrorRate supérieur à 5 % pendant 5 minutes
- CacheHitRate moins de 80 % pendant 30 minutes

Surveillance des performances du MQAR

Lorsque vous utilisez le MQAR, la surveillance vous aide à comprendre comment CloudFront vous choisit entre vos origines et si les scores de qualité répondent aux attentes. Real-time les journaux indiquent ces décisions au fur et à mesure qu'elles sont prises.

Pour surveiller les performances du MQAR

1. Activez les journaux en temps réel pour votre CloudFront distribution.
2. Incluez les champs suivants dans la configuration de votre journal :
 - `r-host`- Le nom d'hôte de l'origine sélectionnée
 - `sr-reason`- La raison du choix de l'origine
 - `x-edge-mqcs`- Le score de confiance en matière de qualité des médias
3. Configurez une destination de journal dans Amazon Kinesis Data Streams ou Amazon Data Firehose.
4. Créez des tableaux de bord ou des alertes sur la base de ces indicateurs pour suivre les scores de qualité et les décisions de sélection de l'origine.

Exemple Exemple de CloudWatch tableau de bord pour la surveillance MQAR

Créez un CloudWatch tableau de bord avec les statistiques suivantes :

- La sélection de l'origine compte par région
- Scores de qualité moyens au fil du temps
- Événements de basculement
- Taux d'erreur 4xx et 5xx par origine

Pour plus d'informations sur la configuration des journaux en temps réel, consultez la section [Real-time relative aux journaux](#) dans le guide du CloudFront développeur.

Dépannage des problèmes courants

Des problèmes peuvent survenir avec votre CloudFront MediaTailor intégration, même avec une planification minutieuse. La compréhension des problèmes courants et des solutions vous permet de les résoudre rapidement et de minimiser l'impact sur les spectateurs.

Si vous rencontrez des problèmes avec votre MediaTailor intégration CloudFront et, consultez les solutions et problèmes courants suivants :

Problèmes de mise en cache des manifestes

Symptôme : manifestes périmés ou publicités non mises à jour

Solution : vérifiez que vous utilisez la politique de `CachingDisabled` cache pour les listes de lecture multivariantes, les listes de lecture multimédia et les chemins MPD. Vérifiez que votre configuration transmet correctement les paramètres de requête.

Erreurs CORS

Symptôme : la console du navigateur affiche des erreurs CORS lors de l'accès au contenu

Solution : configurez une politique d'en-têtes de réponse avec les en-têtes CORS appropriés et associez-la au comportement de votre cache.

MQAR ne fonctionne pas

Symptôme : la sélection de l'origine ne tient pas compte des scores de qualité

Solution : vérifiez que vous avez activé l'option Niveau de qualité multimédia dans les paramètres de votre groupe d'origine. Vérifiez que vous n'utilisez pas de déclencheurs orientés vers l'origine Lambda @Edge.

Échecs d'insertion d'annonces

Symptôme : les publicités ne s'insèrent pas correctement

Solution : vérifiez que votre MediaTailor configuration pointe vers votre CloudFront distribution pour le préfixe du segment de contenu. Vérifiez que votre configuration transmet tous les en-têtes requis.

Pour les problèmes plus complexes, vous pouvez utiliser les approches de dépannage suivantes :

1. Vérifiez les journaux CloudFront de distribution pour détecter les modèles d'erreur
2. Utilisez les outils de développement du navigateur pour inspecter les requêtes réseau
3. Comparez le contenu du manifeste MediaTailor directement par rapport au contenu diffusé via CloudFront
4. Testez avec un lecteur simple qui prend en charge la journalisation détaillée

Pour plus d'assistance en matière de dépannage, consultez la [section](#) Résolution des problèmes du guide de MediaTailor l'utilisateur.

Workflow de dépannage

Suivez cette approche systématique pour résoudre les problèmes liés à votre MediaTailor intégration CloudFront réseau :

Pour résoudre les problèmes CloudFront et les problèmes MediaTailor d'intégration

1. Identifiez le problème spécifique :
 - a. Déterminez si le problème concerne tous les spectateurs ou seulement certains
 - b. Identifiez les types de contenu concernés (manifestes, segments ou les deux)
 - c. Notez les messages d'erreur ou les symptômes
2. Vérifiez CloudFront les journaux :
 - a. Recherchez les codes d'état d'erreur (4xx ou 5xx)
 - b. Vérifiez les hit/miss modèles de cache
 - c. Vérifiez que les demandes sont acheminées vers la bonne origine
3. Vérifiez la configuration :
 - a. Vérifiez les comportements du cache pour des modèles de chemin corrects
 - b. Vérifiez que les politiques de cache sont correctement appliquées
 - c. Vérifiez que les politiques de demande d'origine transmettent les en-têtes nécessaires
4. Testez l'accès direct :
 - a. Essayez d'accéder au contenu directement depuis MediaTailor (en contournant CloudFront)
 - b. Comparez les réponses entre accès direct et CloudFront accès
5. Mettre en œuvre la solution :

- a. Appliquez le correctif approprié en fonction de vos résultats
- b. Testez pour vérifier que le problème est résolu
- c. Documentez le problème et la solution pour référence ultérieure

Étapes suivantes

Après avoir configuré la surveillance et le dépannage pour votre CloudFront MediaTailor intégration réseau, considérez les étapes suivantes :

- Mettez en œuvre un déploiement automatique à l'aide de AWS CloudFormation (voir [Automatisez MediaTailor et créez un CDN avec CloudFormation](#))
- Créez des runbooks pour les scénarios opérationnels courants et les procédures de dépannage
- Mettre en place une solution automatique pour les problèmes courants

Configurer un tiers CDN pour la diffusion MediaTailor des publicités

CDNs Des tiers tels qu'Akamai et Fastly peuvent améliorer de manière significative les performances et l'évolutivité de votre diffusion AWS Elemental MediaTailor publicitaire tout en réduisant les coûts de bande passante. Cependant, la configuration du CDN pour la publicité personnalisée nécessite des paramètres spécifiques différents de ceux de la diffusion vidéo standard.

Pour plus d'informations sur le transfert de paramètres de requête par le biais d'un tiers CDN, consultez [MediaTailor paramètres de requête manifeste](#). Pour les configurations de routage avancées utilisant des variables dynamiques, voir [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

Ce guide vous explique tout au long du processus de configuration de votre CDN tiers pour qu'il fonctionne de manière optimale. MediaTailor Vous allez apprendre à configurer deux comportements essentiels :

- Contournement du manifeste : garantit que chaque spectateur reçoit des insertions publicitaires personnalisées en empêchant la mise en cache du manifeste
- Mise en cache des segments : optimise les performances et réduit les coûts en mettant en cache efficacement le contenu vidéo

Le processus de configuration prend généralement 30 à 60 minutes et inclut la sélection du fournisseur, la configuration, la vérification et l'optimisation. Une fois terminée, vous disposerez d'une configuration CDN qui diffusera des publicités personnalisées de manière efficace tout en garantissant une expérience utilisateur optimale.

Note

Ce guide se concentre sur les configurations d'Akamai et de Fastly. Pour les instructions de CloudFront configuration, voir [intégration d'CloudFront](#).

Conditions préalables

Avant de configurer votre CDN tiers avec MediaTailor, assurez-vous d'avoir :

- Une MediaTailor configuration active qui inclut l'origine de votre contenu et votre serveur de décision publicitaire
- Accès à l'interface de configuration de votre CDN
- Liste des extensions de fichier utilisées dans votre contenu (.m3u8, .mpd, .ts, etc.)
- La documentation de votre fournisseur de CDN à titre de référence

Pour obtenir des instructions de CloudFront configuration plutôt que des instructions tierces CDN, consultez [intégration d'CloudFront](#).

Terminologie

Pour comprendre les exigences de configuration du CDN, vous devez connaître les types de manifestes suivants :

- Manifestes HLS :
 - Liste de lecture multivariée : manifeste de niveau supérieur contenant des liens vers des listes de lecture multimédia
 - Liste de lecture multimédia : le manifeste de deuxième niveau contenant des liens vers des segments de contenu
- DASH manifeste :
 - MPD (Media Presentation Description) : terme standard pour les manifestes DASH

Ce guide fait référence à tous les fichiers manifestes (listes de lecture multivariantes, listes de lecture multimédia, etc. MPDs) collectivement en tant que manifestes lorsqu'il traite des exigences de configuration communes.

Pour connaître les principes généraux de configuration du CDN applicables à tous les fournisseurs, consultez [Configurer l'intégration du CDN avec MediaTailor](#).

Pour obtenir des conseils sur l'optimisation du CDN, consultez [Guide d'optimisation des performances pour le CDN et MediaTailor les intégrations](#).

Configurer les règles de mise en cache du CDN

La configuration de la mise en cache du CDN est essentielle à la diffusion des MediaTailor publicités, car elle détermine la manière dont votre contenu atteint les spectateurs. Une configuration appropriée garantit que les manifestes restent personnalisés pour chaque utilisateur, tandis que les segments sont efficacement mis en cache afin de réduire la charge d'origine et d'améliorer les performances.

Cette configuration prend généralement 15 à 30 minutes par fournisseur de CDN et nécessite deux comportements distincts :

- Gestion des manifestes : empêche la mise en cache pour garantir que chaque spectateur reçoit des insertions publicitaires personnalisées
- Mise en cache des segments : optimise l'efficacité du cache pour le contenu vidéo afin d'améliorer les performances de diffusion

Suivez ces étapes pour configurer les règles de mise en cache de votre CDN pour une diffusion publicitaire optimale.

Choisissez votre fournisseur de CDN dans les onglets suivants pour obtenir des instructions spécifiques :

Akamai

Configurez ces deux comportements dans votre propriété Akamai :

- Gestion des manifestes pour empêcher la mise en cache
- Mise en cache des segments pour des performances optimales

Configuration de la livraison du manifeste

Configurez votre CDN Akamai pour éviter de mettre en cache des manifestes afin que chaque utilisateur reçoive des publicités personnalisées.

Les fichiers manifestes contiennent les points d'insertion d'annonces personnalisés MediaTailor générés pour chaque spectateur. La mise en cache de ces fichiers permettrait à tous les spectateurs de voir des publicités identiques, ce qui irait à l'encontre de l'objectif de la publicité personnalisée.

Procédez comme suit pour les demandes de manifeste (fichiers se terminant par .m3u8, .mpd ou .smil) :

1. Créez un comportement correspondant aux extensions de fichier manifeste (.m3u8, .mpd, .smil)
2. Définir l'option de mise en cache sur No Store
3. Configurer les clés de cache pour inclure tous les paramètres de requête
4. Activez l'en-tête Forward Host pour un routage d'origine correct
5. Configurez le transfert d'en-têtes pour tous les en-têtes. Pour les exigences minimales, voir [En-têtes requis pour l'intégration du MediaTailor CDN](#).

Configuration de la diffusion par segment

Configurez votre CDN Akamai pour mettre en cache les segments vidéo afin d'optimiser l'efficacité du CDN et de réduire la charge d'origine.

Les segments vidéo sont des fichiers de contenu réels qui peuvent être mis en cache en toute sécurité car ils ne contiennent pas d'informations personnalisées. Une mise en cache appropriée des segments réduit les coûts de bande passante et améliore les performances de lecture pour les spectateurs.

Procédez comme suit pour les demandes de segment (fichiers se terminant par .ts, .mp4, .m4s, etc.) :

1. Créez un comportement correspondant aux extensions de fichiers de segment (.ts, .mp4, .m4s)
2. Réglez le contrôle du cache Honor Origin sur Oui
3. Configurez les paramètres par défaut time-to-live (TTL) lorsque les en-têtes d'origine sont manquants :
 - TTL par défaut : 86400 secondes (24 heures)

- TTL maximum : 604800 secondes (7 jours)

 Note

Après avoir configuré ces comportements, activez les modifications de vos propriétés dans le centre de contrôle Akamai.
Les modifications prennent effet après activation.

Fastly

Créez ces deux configurations dans votre service Fastly :

- Gestion des manifestes pour empêcher la mise en cache
- Mise en cache des segments pour des performances optimales

Configuration de la livraison du manifeste

Configurez votre CDN Fastly pour contourner la mise en cache des fichiers manifestes afin que chaque spectateur reçoive un contenu publicitaire personnalisé.

Les fichiers manifestes doivent être accessibles MediaTailor pour chaque demande afin de garantir une personnalisation appropriée des annonces. Le contournement du cache pour ces fichiers garantit que les paramètres de ciblage uniques de chaque utilisateur sont traités correctement.

Pour les demandes de manifeste, procédez comme suit :

1. Création d'une condition de demande pour identifier les chemins du manifeste
2. Définissez la condition de cache sur Ne pas mettre en cache pour ces demandes
3. Configurer les paramètres de transfert pour inclure tous les paramètres de requête
4. Ajoutez User-Agent à votre configuration de transfert d'en-têtes

Configuration de la diffusion par segment

Configurez votre CDN Fastly pour mettre en cache les segments vidéo afin d'améliorer les performances de diffusion et de réduire le trafic d'origine.

La mise en cache des segments est essentielle pour une diffusion rentable et une expérience utilisateur optimale. Ces fichiers sont identiques pour tous les utilisateurs et bénéficient considérablement de la mise en cache du CDN.

Pour les demandes de segment, procédez comme suit :

1. Création d'une condition de demande pour identifier les chemins de segment
2. Définissez les paramètres du cache pour honorer les en-têtes du cache d'origine
3. Configurer la valeur par défaut time-to-live (TTL) à 86 400 secondes (24 heures) lorsque les en-têtes d'origine sont manquants

 Note

Après avoir effectué ces modifications, activez une nouvelle version de votre service Fastly.
La configuration prend effet après activation.

Vérifiez la configuration de votre CDN

La vérification garantit que la configuration de votre CDN fonctionne correctement avant que vous n'y dirigiez le trafic de production. Ces tests confirment que la personnalisation des publicités fonctionne correctement et que la mise en cache apporte les avantages escomptés en termes de performances.

La vérification complète prend généralement 10 à 15 minutes et doit être effectuée à partir de plusieurs emplacements géographiques si possible.

Après avoir configuré votre CDN, effectuez les vérifications suivantes pour vérifier qu'il fonctionne correctement :

1. Personnalisation du manifeste de test :
 - a. Demandez la même URL de contenu avec des paramètres publicitaires différents
 - b. Vérifiez que chaque demande renvoie des insertions publicitaires différentes
2. Tester la mise en cache des segments :
 - a. Vérifiez les métriques du CDN pour connaître le taux de réussite du cache de segments (il doit être supérieur à 90 %)
 - b. Surveillez le trafic d'origine pour confirmer qu'il est inférieur à celui de la livraison directe

3. Testez les performances de lecture :

- a. Diffusez du contenu via votre CDN à partir de différents endroits
- b. Vérifiez la fluidité de la lecture sans aucun problème de mise en mémoire tampon

Pour des méthodologies de test complètes et des procédures de validation avancées, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#).

Optimisez les performances du CDN

Après avoir vérifié votre configuration de base, implémentez ces optimisations pour optimiser les performances et minimiser les coûts :

Surveillez les indicateurs de performance clés

Suivez ces indicateurs pour garantir des performances optimales :

Taux de réussite du cache

Objectif : plus de 90 % pour les segments vidéo

Conséquence : des ratios plus élevés réduisent la charge d'origine et améliorent l'expérience utilisateur

Surveillance : consultez quotidiennement le tableau de bord d'analyse de votre fournisseur de CDN

Temps de réponse d'origine

Objectif : moins de 200 ms pour les demandes de manifeste

Conséquence : l'accélération de la diffusion des manifestes réduit le temps de démarrage pour les utilisateurs

Surveillance : configurez des alertes pour des temps de réponse supérieurs à 500 ms

Taux d'erreur

Objectif : Moins de 0,1 % pour toutes les demandes

Conséquence : les taux d'erreur élevés indiquent des problèmes de configuration ou d'origine.

Surveillance : configurez des alertes pour les taux d'erreur supérieurs à 1 %

Affiner le comportement de mise en cache

Ajustez ces paramètres en fonction des caractéristiques de votre contenu et des habitudes d'audience :

Optimisation TTL des segments

Contenu en direct : utilisez un TTL plus court (1 à 4 heures) pour garantir des mises à jour en temps opportun

Contenu VOD : utilisez un TTL plus long (24 à 48 heures) pour optimiser l'efficacité du cache

Segments publicitaires : envisagez une durée TTL plus courte (30 minutes à 2 heures) pour le contenu publicitaire fréquemment mis à jour

Pour des recommandations TTL complètes et des stratégies de mise en cache pour tous les MediaTailor flux de travail, voir. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

Optimisation géographique

Origines multirégionales : configurez la sélection de l'origine en fonction de l'emplacement du spectateur

Emplacements périphériques : activez des emplacements périphériques supplémentaires dans les régions à forte concentration de spectateurs

Failover : configurer les origines des sauvegardes pour une haute disponibilité

Optimisez les coûts

Mettez en œuvre ces stratégies pour réduire les coûts du CDN tout en maintenant les performances :

- Compression : activez la compression gzip pour les fichiers manifestes afin de réduire l'utilisation de la bande passante
- Stratégie de purge : mise en œuvre d'une purge sélective du cache au lieu d'une purge complète du cache

- Analyse du trafic : passez en revue les modèles de trafic tous les mois pour identifier les opportunités d'optimisation
- Sélection du niveau : utilisez les niveaux de service CDN appropriés en fonction de vos exigences de performance

Résoudre les problèmes liés aux CDN tiers

Les problèmes de configuration du CDN se manifestent généralement par des problèmes de personnalisation des publicités ou par une dégradation des performances. Utilisez cette approche systématique pour identifier et résoudre les problèmes les plus courants qui affectent la diffusion des MediaTailor publicités.

La plupart des dépannages peuvent être effectués en 15 à 30 minutes en vérifiant les symptômes spécifiques et en appliquant les solutions correspondantes.

Si les spectateurs rencontrent des problèmes liés à la diffusion des publicités ou à la qualité de lecture, utilisez ce guide pour identifier et résoudre les problèmes courants de configuration du CDN :

Akamai : manifestes mis en cache

Symptôme : les spectateurs voient des publicités identiques même lorsque vous configurez des paramètres de ciblage différents.

Solution : Vérifiez que vous avez appliqué l'option de mise en cache No Store aux chemins du manifeste.

Vérifiez également que vous avez inclus les paramètres de requête dans la clé de cache.

Rapidement : clés de cache incorrectes

Symptôme : les spectateurs sont confrontés à une personnalisation incohérente des publicités.

Les spectateurs peuvent également voir des publicités destinées à d'autres spectateurs.

Solution : Vérifiez que vous avez configuré les paramètres de transfert pour inclure tous les paramètres de requête dans la clé de cache.

Généralités : trafic d'origine élevé

Symptôme : le trafic de vos serveurs d'origine est étonnamment élevé

Solution : vérifiez les paramètres de mise en cache des segments et les time-to-live valeurs (TTL).

Vérifiez les taux de réussite du cache dans les statistiques de votre CDN.

Généralités : erreurs de lecture

Symptôme : les spectateurs rencontrent des problèmes de mise en mémoire tampon ou de lecture

Solution : vérifiez les règles de routage du CDN et l'état de l'origine.

Vérifiez que tous les en-têtes requis sont correctement transférés.

 Note

Si ces solutions ne résolvent pas votre problème, consultez la documentation de votre fournisseur de CDN. Vous pouvez également contacter leur équipe d'assistance pour des étapes de dépannage supplémentaires.

Pour obtenir des conseils généraux de résolution des problèmes liés au CDN, consultez [Résoudre les problèmes liés à l'intégration MediaTailor du CDN](#).

Guide d'optimisation des performances pour le CDN et MediaTailor les intégrations

AWS Elemental MediaTailor les performances peuvent être maximisées grâce à l'optimisation systématique du réseau de diffusion de contenu (CDN). Que vous mettiez en œuvre l'insertion publicitaire côté serveur (SSAI), l'assemblage de canaux ou des flux de travail combinés, les principes d'optimisation et les objectifs de performance restent cohérents. Ce guide fournit des techniques d'optimisation complètes et des points de référence qui s'appliquent à toutes les MediaTailor implémentations.

Pour une optimisation avancée du routage à l'aide de variables dynamiques et d'alias de configuration, consultez [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#). Pour les stratégies d'optimisation des paramètres de requête, voir [MediaTailor paramètres de requête manifeste](#).

Présentation du flux de travail d'optimisation :

1. Configuration de la mise en cache : définissez les valeurs TTL et les comportements de cache appropriés
2. Optimisation du routage : configurez le routage des demandes et les politiques d'origine
3. Mesurer les performances - Effectuer le suivi par rapport aux points de référence établis
4. Appliquer des techniques avancées - Implémenter des fonctionnalités d'optimisation supplémentaires

Rubriques

- [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
- [Optimisation du routage des demandes pour le CDN et les MediaTailor intégrations](#)
- [Benchmarks de performance pour le CDN et les intégrations MediaTailor](#)
- [Techniques d'optimisation avancées pour le CDN et MediaTailor les intégrations](#)

Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor les exigences de mise en cache varient selon le type de flux de travail et le format de contenu. Une configuration de mise en cache appropriée est essentielle pour des performances, une rentabilité et une expérience utilisateur optimales. Les sections suivantes fournissent des conseils de mise en cache détaillés pour les différentes MediaTailor implémentations.

Mise en cache d'insertion de publicités côté serveur (SSAI)

Pour les flux de travail d'insertion d'annonces côté serveur, les manifestes personnalisés ne peuvent pas être mis en cache, mais le contenu et les segments publicitaires doivent être mis en cache de manière agressive :

Paramètres de mise en cache SSAI CDN

Type de contenu	TTL	Modèle de chemin	Éléments clés du cache
Listes de lecture multivariantes	0 secondes	/v1/maître/ *	Chemin d'URL + tous les paramètres de requête
Listes de lecture multimédia	0 secondes	/v1/manifest/ *	Chemin d'URL + tous les paramètres de requête

Type de contenu	TTL	Modèle de chemin	Éléments clés du cache
TABLEAU DE BORD MPDs	0 secondes	/v1/dash/ *	Chemin d'URL + tous les paramètres de requête
Segments de contenu	24 heures et plus	Chemins spécifiques au contenu	Chemin d'URL uniquement
Segments publicitaires	24 heures et plus	/v1/segment/ *	Chemin d'URL uniquement

- Définissez un TTL de 0 seconde pour les manifestes personnalisés afin de garantir que les spectateurs reçoivent du contenu up-to-date publicitaire
- Configurez des valeurs TTL plus longues pour le contenu et les segments publicitaires afin d'optimiser l'efficacité du cache
- Configurez des comportements de cache qui incluent des paramètres de personnalisation dans la clé de cache si vous soutenez la publicité ciblée
- Mettre en œuvre le regroupement des demandes au niveau du CDN pour gérer efficacement les demandes simultanées

Paramètres de configuration TTL recommandés

Pour des performances SSAI optimales, configurez vos politiques de cache CDN avec ces paramètres TTL spécifiques :

Paramètres de configuration TTL SSAI

Type de contenu	Réglage TTL	Valeur recommandée
Segments publicitaires	TTL minimum	1 seconde
Segments publicitaires	TTL maximal	86400 secondes (24 heures)
Segments publicitaires	TTL par défaut	86400 secondes (24 heures)
Segments de contenu	TTL minimum	1 seconde
Segments de contenu	TTL maximal	86400 secondes (24 heures)

Type de contenu	Réglage TTL	Valeur recommandée
Segments de contenu	TTL par défaut	86400 secondes (24 heures)

Ces paramètres garantissent :

- TTL minimum de 1 seconde : permet une invalidation rapide du cache en cas de besoin tout en évitant les demandes d'origine excessives
- TTL maximum de 24 heures : équilibre l'efficacité du cache avec les exigences de fraîcheur du contenu
- TTL par défaut de 24 heures : fournit une mise en cache optimale pour les segments qui n'ont pas d'en-têtes de contrôle de cache explicites

Mise en cache d'insertion de publicités guidée par le serveur (SGAI)

L'insertion publicitaire guidée par le serveur (SGAI) permet une mise en cache efficace du CDN grâce à des manifestes multimédias pouvant être mis en cache qui utilisent des modèles d'URL prévisibles. Cette section se concentre sur les exigences de configuration spécifiques au CDN pour des performances SGAi optimales.

Configuration de la mise en cache du CDN pour SGAi

Configurez votre CDN avec ces comportements de mise en cache spécifiques à SGAi :

Paramètres de mise en cache du CDN SGAi

Type de contenu	TTL	Modèle de chemin	Éléments clés du cache
Listes de lecture multivariantes SGAi (ne pas mettre en cache)	0 secondes (ne pas mettre en cache)	/v1/maître/ *	Chemin d'URL + paramètres de requête sélectionnés
Listes de lecture multimédia SGAi	1 à 4 secondes (durée d'un demi-segment)	/v1/i-média/ *	Chemin d'URL + paramètres de requête sélectionnés

Type de contenu	TTL	Modèle de chemin	Éléments clés du cache
Réponses aux listes d'actifs (ne pas mettre en cache)	0 secondes (ne pas mettre en cache)	/v1/interstitiels/ *	Chemin d'URL + tous les paramètres de requête
Segments publicitaires	24 heures et plus	Chemins spécifiques à l'annonce	Chemin d'URL uniquement

Configuration du comportement du cache

Configurez des comportements de cache dédiés pour le contenu SGAI :

- Comportement du manifeste SGAI - Créez un comportement de cache pour les `/v1/i-media/*` chemins avec un TTL de 1 à 4 secondes
- Comportement de la liste d'actifs - Créez un comportement de cache pour les `/v1/interstitiels/*` chemins avec un TTL de 0 seconde
- Gestion des paramètres de requête : incluez uniquement les paramètres de ciblage essentiels dans les clés de cache pour optimiser l'efficacité du cache
- En-têtes de demande Origin - Transférez les en-têtes nécessaires au ciblage publicitaire tout en préservant la capacité de mise en cache

Mise en cache des assemblages de canaux

Pour les flux de travail d'assemblage de canaux, les manifestes peuvent être mis en cache pendant de courtes périodes, tandis que les segments doivent être mis en cache de manière agressive :

Paramètres de mise en cache du CDN d'assemblage de canaux

Type de contenu	VOD TTL	TTL en direct	Modèle de chemin	Éléments clés du cache
Listes de lecture multivariantes	5 à 30 minutes	5 à 10 secondes	Chemins spécifiques au canal	Chemin d'URL + tous les paramètres de requête

Type de contenu	VOD TTL	TTL en direct	Modèle de chemin	Éléments clés du cache
Listes de lecture multimédia	5 à 30 minutes	2 à 5 secondes	Chemins spécifiques au canal	Chemin d'URL + tous les paramètres de requête
TABLEAU DE BORD MPDs	5 à 30 minutes	5 à 10 secondes	Chemins spécifiques au canal	Chemin d'URL + tous les paramètres de requête
Segments de contenu	24 heures et plus	5 à 15 minutes	Chemins spécifiques au contenu	Chemin d'URL uniquement
Segments publicitaires	24 heures et plus	24 heures et plus	Chemins spécifiques à l'annonce	Chemin d'URL uniquement

- Définissez des valeurs TTL courtes pour les manifestes afin de garantir que les spectateurs reçoivent up-to-date la programmation
- Configurez des valeurs TTL plus longues pour les segments de contenu afin d'optimiser l'efficacité du cache
- Configurez des comportements de cache qui incluent des paramètres de décalage temporel dans la clé de cache si vous acceptez l'affichage décalé dans le temps
- Incluez les paramètres de requête dans la clé de cache pour gérer correctement les demandes de visualisation décalées dans le temps

Pour connaître les paramètres de configuration TTL détaillés et les meilleures pratiques, consultez [the section called “Optimisation de la mise en cache”](#).

Mise en cache combinée du SSAI et de l'assemblage des canaux

Lorsque vous implémentez à la fois l'assemblage des canaux et le SSAI, assurez-vous que votre stratégie de mise en cache est cohérente pour les deux services afin d'éviter les conflits et d'optimiser les performances :

Comparaison des paramètres de mise en cache des flux de travail combinés

Type de contenu	Assemblage du canal	SSAI	Recommandation combinée
Manifestes de VOD	5 à 30 minutes	0 secondes	(utilisez une configuration séparée)
Manifestes en direct	2 à 10 secondes	0 secondes	(utilisez une configuration séparée)
Manifestes VOD SGAI	5 à 30 minutes	5 à 30 minutes	5 à 30 minutes (manifestes pouvant être mis en cache)
Manifestes SGAI Live	2 à 4 secondes	2 à 4 secondes	2 à 4 secondes (manifestes pouvant être mis en cache)
Segments de contenu	24 heures et plus	24 heures et plus	24 heures et plus (constant)
Segments publicitaires	24 heures et plus	24 heures et plus	24 heures et plus (constant)

Cette configuration maximise l'efficacité du cache tout en garantissant que les spectateurs reçoivent des up-to-date manifestes pour une insertion publicitaire personnalisée.

Optimisation du routage des demandes pour le CDN et les MediaTailor intégrations

Implémentez ces optimisations de routage pour toutes les intégrations de AWS Elemental MediaTailor CDN :

- Créez des comportements de cache distincts pour les demandes de manifeste et de segment
- Configurer les politiques de demande d'origine pour contrôler le transfert d'en-têtes
- Mettre en place des mécanismes appropriés de gestion des erreurs et de basculement
- Implémentez des boucliers d'origine, s'ils sont disponibles dans votre CDN, afin de réduire la charge d'origine

- Mettre en œuvre le regroupement des demandes au niveau du CDN pour gérer efficacement les demandes simultanées

Benchmarks de performance pour le CDN et les intégrations MediaTailor

Lorsque vous optimisez l'intégration de votre AWS Elemental MediaTailor CDN, visez ces tests de performance. Ces objectifs s'appliquent à toutes les MediaTailor implémentations, y compris le SSAI, l'assemblage de canaux et les flux de travail combinés :

Objectifs relatifs au taux de réussite du cache

Segments de contenu : taux d'accès au cache supérieur à 95 %

Segments publicitaires : taux de réussite du cache supérieur à 90 %

Manifestes : Non applicable pour le SSAI (ne doivent pas être mis en cache pour l'insertion d'annonces personnalisées) ; 85 % ou plus pour l'assemblage des chaînes

Repères de latence

Latence des demandes de manifeste : moins de 100 ms (P95)

Diffusion du segment de contenu : moins de 50 ms (P95)

Diffusion du segment publicitaire : moins de 75 ms (P95)

End-to-end temps de démarrage : moins de 2 secondes

Métriques de charge d'origine

Demandes d'origine par téléspectateur : moins de 0,1 demande par minute et par téléspectateur

Bande passante d'origine par spectateur : moins de 5 % de la bande passante totale du spectateur

Objectifs en matière de taux d'erreur

Erreurs manifestes : moins de 0,1 %

Erreurs de segment : moins de 0,01 %

Rebuffering déclaré par le joueur : moins de 1 %

Benchmarks d'évolutivité

Support d'un trafic 10 fois supérieur à la normale pendant les périodes de pointe, sans dégradation

Capacité à traiter plus de 1 000 demandes par seconde et par canal

Utilisez CloudWatch les métriques Amazon pour suivre ces indicateurs de performance. Pour des instructions de surveillance détaillées, voir [Configuration des outils de surveillance](#).

Techniques d'optimisation avancées pour le CDN et MediaTailor les intégrations

Après avoir mis en œuvre les optimisations de base de la mise en cache et du routage, considérez les techniques avancées suivantes pour améliorer encore les performances :

Implémentation d'Origin Shield

Origin Shield ajoute une couche de mise en cache entre les emplacements périphériques du CDN et votre serveur d'origine, réduisant ainsi la charge d'origine et améliorant les performances :

- Activez Origin Shield pour les contenus à fort trafic et la diffusion en direct
- Choisissez des agences Origin Shield proches de vos MediaTailor régions
- Surveillez les taux de réussite du cache Origin Shield et ajustez-les si nécessaire
- Envisagez plusieurs sites Origin Shield pour les déploiements internationaux

Optimisation de la compression du contenu

Optimisez la compression du contenu pour réduire la bande passante et améliorer la vitesse de diffusion :

- Activer la compression gzip pour les fichiers manifestes
- Configurer le transfert d'en-tête Accept-Encoding pour MediaTailor la compression du manifeste
- Utilisez la compression Brotli lorsqu'elle est prise en charge pour économiser davantage de bande passante
- Évitez de compresser des segments vidéo déjà compressés

Stratégies d'optimisation régionales

Optimisez les performances pour le public mondial grâce à des stratégies régionales :

- Déployez des MediaTailor configurations dans plusieurs régions pour un public mondial
- Utilisez le géo-routage pour diriger les spectateurs vers la région la plus proche MediaTailor
- Configurer le basculement régional pour une haute disponibilité
- Surveillez les indicateurs de performance régionaux séparément

Considérations relatives à l'architecture pour le CDN et les MediaTailor intégrations

Positionnez correctement votre réseau de diffusion de contenu (CDN) dans votre architecture afin de garantir des performances et une fiabilité optimales avec AWS Elemental MediaTailor. L'architecture recommandée place le CDN entre les spectateurs et MediaTailor non entre MediaTailor votre ordinateur d'origine.

Pour obtenir des diagrammes d'architecture détaillés et des explications sur le flux de travail, consultez les rubriques suivantes.

- [Insertion d'annonces avec CDN](#) pour les diagrammes d'architecture d'insertion de publicités et le flux de travail détaillé
- [Comprendre l'architecture du CDN](#) pour l'assemblage des canaux, les schémas d'architecture et le flux de travail

Positionnez correctement votre CDN dans votre architecture :

1. Placez votre CDN entre les joueurs et MediaTailor (pas entre MediaTailor et votre origine).

Cette architecture permet à votre CDN de mettre en cache des segments publicitaires et des segments de contenu. Dans le même temps, MediaTailor peut générer des manifestes personnalisés pour chaque spectateur.

2. Créez des comportements de cache distincts pour les différents types de demandes :

- Demandes de manifeste (pas de mise en cache)
- Segments de contenu (TTL plus long)
- Segments publicitaires (TTL plus long)

3. Configurez la gestion appropriée des erreurs :

- Configurez une mise en cache négative (stockage temporaire des réponses aux erreurs) pour éviter de surcharger votre source de demandes répétées lors d'interruptions de service. La mise en cache négative signifie que le CDN stockera temporairement les réponses aux erreurs (comme les erreurs 404 ou 500) afin d'éviter les demandes répétées de contenu qui n'existe pas ou qui est temporairement indisponible.
- Configurer les codes de réponse aux erreurs appropriés et le comportement des nouvelles tentatives

4. Implémenter la mise en cache intermédiaire (bouclier d'origine) :

Origin Shield est une fonctionnalité qui crée une couche de mise en cache supplémentaire entre les emplacements périphériques du CDN et votre serveur d'origine. Cela réduit le nombre de demandes redondantes qui atteignent votre serveur d'origine.

- Configurer une couche de mise en cache intermédiaire entre les emplacements périphériques et votre origine
- Réduisez le nombre de demandes redondantes adressées à votre source en cas d'échec du cache
- Améliorez les taux de réussite du cache sur l'ensemble de votre infrastructure CDN

Fonctionnalités CDN avancées pour MediaTailor

Après avoir implémenté la configuration de base du réseau de diffusion de contenu (CDN), explorez ces fonctionnalités avancées pour améliorer encore les performances et la fiabilité de votre plateforme de AWS Elemental MediaTailor streaming.

Routage adapté à la qualité des médias (MQAR)

La MQAR est une CloudFront fonctionnalité d'Amazon qui sélectionne automatiquement la source de contenu de la plus haute qualité en fonction des indicateurs de performance du réseau en temps réel. Au lieu d'utiliser un serveur d'origine fixe, MQAR achemine les demandes de manière dynamique vers l'origine la plus performante en fonction de facteurs tels que la latence et le débit. Cela permet de garantir que les spectateurs reçoivent un flux de la meilleure qualité possible, même en cas de fluctuations du réseau.

Si vous utilisez Amazon CloudFront, implémentez MQAR pour sélectionner automatiquement la source de contenu de la plus haute qualité en fonction de statistiques en temps réel. Pour plus de détails, consultez [intégration d'CloudFront](#) la section sur CloudFront l'intégration.

Filtrage des manifestes

Le filtrage des manifestes vous permet de personnaliser les rendus (versions de qualité différente du même contenu) qui sont inclus dans les manifestes destinés MediaTailor aux spectateurs. Le filtrage permet d'optimiser l'utilisation de la bande passante en supprimant les rendus qui ne sont pas adaptés à certains appareils ou à certaines conditions du réseau. Par exemple, vous pouvez supprimer les rendus 4K pour les appareils mobiles ou les connexions à faible bande passante.

Pour plus d'informations sur la mise en œuvre du filtrage des manifestes avec AWS Elemental MediaPackage, consultez [MediaPackage Intégration au CDN](#).

Stratégie multi-CDN

Une stratégie multi-CDN utilise plusieurs fournisseurs de CDN simultanément pour améliorer la fiabilité et les performances. Si un CDN rencontre des problèmes, le trafic peut automatiquement être transféré vers un autre fournisseur. Cette approche est particulièrement utile pour les événements en direct très médiatisés où la fiabilité est essentielle.

Pour plus d'informations sur la mise en œuvre d'une stratégie multi-CDN avec MediaTailor, voir [Planifier l'intégration du CDN](#)

Surveillez les opérations et les performances du MediaTailor CDN

Une surveillance efficace de votre intégration AWS Elemental MediaTailor et de celle de votre réseau de diffusion de contenu (CDN) garantit une diffusion de contenu fiable, des performances optimales et une détection rapide des problèmes. Cette approche de surveillance s'applique à toutes les MediaTailor implémentations, y compris l'insertion de publicités côté serveur (SSAI), l'assemblage de canaux et les flux de travail combinés.

Le suivi de l'intégration de votre CDN vous permet de :

- Détectez et résolvez les problèmes avant qu'ils n'affectent les spectateurs
- Suivez les indicateurs de performance clés et maintenez la qualité du service
- Maintenir des performances de cache optimales et réduire la charge d'origine
- Assurez-vous que les taux de réussite de l'insertion d'annonces répondent aux exigences de l'entreprise

Pour résoudre les problèmes liés aux paramètres qui peuvent apparaître dans les données de surveillance, voir [MediaTailor guide de dépannage des paramètres](#) Pour plus d'informations sur la

surveillance de l'utilisation des paramètres de requête, consultez [MediaTailor paramètres de requête manifeste](#).

Indicateurs de performance essentiels du CDN

Suivez ces indicateurs de base pour évaluer l'efficacité de votre CDN lors des MediaTailor mises en œuvre :

Taux de réussite du cache

Ce qu'il mesure : le pourcentage de demandes traitées depuis le cache du CDN par rapport à l'origine.

Valeurs cibles :

- Segments de contenu : taux d'accès au cache de 95 % ou plus
- Segments publicitaires : taux d'accès au cache de 90 % ou plus
- Manifestes : varie selon l'implémentation (les manifestes personnalisés ne doivent pas être mis en cache)

Pourquoi c'est important : des taux d'accès au cache plus élevés réduisent la charge d'origine, améliorent les temps de réponse et réduisent les coûts de bande passante.

Volume de demandes d'origine

Ce qu'il mesure : le nombre de demandes parvenant à vos serveurs MediaTailor d'origine.

Modèle cible : devrait rester faible et stable, avec des pics occasionnels dus à des erreurs de cache ou à de nouveaux contenus.

Pourquoi c'est important : les volumes élevés de demandes d'origine indiquent une inefficacité de la mise en cache et peuvent avoir un impact sur MediaTailor les performances.

Latence de réponse

Ce qu'il mesure : la rapidité avec laquelle le CDN répond aux demandes des utilisateurs.

Valeurs cibles :

- Contenu mis en cache : moins de 100 ms (P95)
- Demandes d'origine : moins de 500 ms (P95)

- Demandes de manifeste : moins de 100 ms (P95)
- Demandes de segment : moins de 50 ms (P95)

Pourquoi c'est important : la faible latence garantit une lecture fluide et une bonne expérience utilisateur.

Taux d'erreur

Ce qu'il mesure : le pourcentage de demandes qui génèrent des erreurs HTTP.

Valeurs cibles :

- Erreurs 4xx : moins de 0,1 % du total des demandes
- Erreurs 5xx : moins de 0,01 % du total des demandes
- Erreurs d'origine : moins de 0,05 % des demandes d'origine

Pourquoi c'est important : les taux d'erreur élevés indiquent des problèmes de configuration ou de service qui ont une incidence sur l'expérience utilisateur.

MediaTailor métriques spécifiques

Surveillez ces MediaTailor indicateurs parallèlement aux indicateurs du CDN pour une visibilité complète :

Taux de remplissage des publicités

Indicateurs clés : `Avail.FillRate` et `AdDecisionServer.FillRate`

Valeurs cibles : supérieures à 90 % pour les deux mesures

Pourquoi c'est important : impacte directement les recettes publicitaires et l'expérience des spectateurs

Performance de génération de manifestes

Indicateurs clés : `GetManifest.Latency` et `GetManifest.Errors`

Valeurs cibles : latence inférieure à 200 ms, taux d'erreur inférieur à 1 %

Pourquoi c'est important : affecte le temps de démarrage et la fiabilité de la lecture

État du serveur Ad Decision

Indicateurs clés : `AdDecisionServer.LatencyAdDecisionServer.Errors`, et `AdDecisionServer.Timeouts`

Valeurs cibles : latence inférieure à 1 000 ms, taux d'erreur inférieur à 5 %, délais d'attente minimaux

Pourquoi c'est important : les performances de l'ADS ont un impact direct sur le succès de l'insertion des annonces

Configuration des outils de surveillance

Configurez ces outils pour surveiller efficacement votre intégration MediaTailor et celle du CDN :

CloudWatch Intégration avec Amazon

Amazon CloudWatch fournit les bases de la surveillance de votre intégration MediaTailor et de celle du CDN :

MediaTailor métriques

MediaTailor publie automatiquement des statistiques CloudWatch permettant de suivre les demandes, les réponses et les erreurs. Les métriques clés incluent :

- `RequestCount`: nombre total de demandes adressées à MediaTailor
- `ResponseTime`: latence MediaTailor de réponse
- `4xxErrorCount` et `5xxErrorCount` : Suivi des erreurs

Pour une liste complète des MediaTailor indicateurs, voir [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#).

Métriques du CDN

Activez la collecte de métriques CDN en CloudWatch incluant :

- Taux de réussite du cache pour différents types de contenu
- Nombre de demandes d'origine et temps de réponse
- Taux d'erreur par code d'état

Pour CloudFront des statistiques spécifiques, activez le suivi détaillé dans vos paramètres de distribution.

Configuration du tableau de bord

Créez des tableaux de bord qui offrent une visibilité sur vos performances MediaTailor et celles du CDN :

1. Créez un tableau de bord unifié : combinez MediaTailor les métriques du CDN dans un seul CloudWatch tableau de bord pour une visibilité complète.
2. Organiser par flux de travail : regroupez les métriques par type d'implémentation (SSAI, assemblage de canaux ou flux de travail combinés).
3. Incluez les indicateurs de performance clés :
 - Tendances du taux de réussite du cache au fil du temps
 - Percentiles de latence de réponse (P50, P95, P99)
 - Tendances et pics du taux d'erreur

Pour obtenir des conseils détaillés sur la création de tableaux de bord, consultez la section [CloudWatchTableaux de bord](#).

Configuration des alertes

Configurez des alertes pour détecter les problèmes avant qu'ils n'affectent les utilisateurs :

Alertes critiques

Configurez des alertes immédiates en cas de problèmes graves :

- Taux d'erreur élevés : alerte lorsque le nombre d'erreurs 5xx dépasse 0,1 % des demandes sur une période de 5 minutes
- Problèmes avec le serveur d'origine : alerte lorsque le temps de réponse d'origine dépasse 2 secondes
- Diminution du taux d'accès au cache : alerte lorsque le taux d'accès au cache tombe en dessous de 70 % pour les manifestes ou de 85 % pour les segments

Alertes d'avertissement

Configurez des alertes d'alerte précoce en cas de dégradation des performances :

- La latence augmente : alerte lorsque le temps de réponse du P95 dépasse 200 ms
- Diminution de l'efficacité du cache : alerte lorsque le taux d'accès au cache tombe en dessous de 90 % pour les segments

Liste de contrôle de mise en œuvre

Utilisez cette liste de contrôle pour garantir une couverture complète de la surveillance :

1. Collecte de métriques :

- ✓ MediaTailor métriques activées dans CloudWatch
- ✓ Surveillance détaillée du CDN activée

2. Configuration du tableau de bord :

- ✓ Tableau de CloudWatch bord unifié créé
- ✓ Indicateurs clés visualisés avec des plages de temps appropriées
- ✓ Tableau de bord partagé avec les équipes concernées

3. Configuration des alertes :

- ✓ Alertes critiques configurées avec notification immédiate
- ✓ Alertes d'avertissement configurées pour une détection précoce
- ✓ Procédures d'escalade des alertes documentées

4. Procédures opérationnelles :

- ✓ Procédures de réponse aux incidents documentées
- ✓ Un calendrier d'examen régulier a été établi
- ✓ Entraînement de l'équipe terminé

Rubriques en relation

Pour obtenir des conseils supplémentaires sur des scénarios de surveillance spécifiques :

- Optimisation des performances : pour des techniques d'optimisation détaillées basées sur les données de surveillance, voir [Optimisation des performances du CDN](#).
- Résolution des problèmes : pour connaître les procédures de dépannage détaillées utilisant les données de surveillance, consultez la documentation de dépannage spécifique à votre flux de travail.

- Analyse des journaux : pour une analyse et une surveillance complètes des journaux, voir [Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor](#).

Tests et validation pour le CDN et MediaTailor les intégrations

AWS Elemental MediaTailor L'intégration du réseau de diffusion de contenu (CDN) nécessite des tests approfondis pour garantir une diffusion publicitaire fiable et des performances optimales. Les tests systématiques permettent d'identifier les problèmes d'intégration avant qu'ils n'affectent les utilisateurs. Utilisez cette approche globale pour valider votre réseau de diffusion de contenu et MediaTailor son intégration dans différents scénarios, configurations et conditions de charge.

Sujets connexes :

- Pour la configuration opérationnelle, y compris la surveillance et la journalisation, voir [Surveillance du CDN](#)
- Pour résoudre les problèmes en cas de problème, voir [Résoudre les problèmes liés à l'intégration du CDN](#)
- Pour obtenir des conseils sur l'optimisation des performances, voir [Guide d'optimisation des performances pour le CDN et MediaTailor les intégrations](#)

Rubriques

- [Prérequis de test et configuration pour le CDN et les intégrations MediaTailor](#)
- [Méthodologie de test systématique pour le CDN et les intégrations MediaTailor](#)
- [Liste de contrôle des tests préalables au déploiement pour le CDN et les intégrations MediaTailor](#)
- [Référence des outils et utilitaires de test pour le CDN et MediaTailor les intégrations](#)

Prérequis de test et configuration pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor les tests d'intégration du réseau de diffusion de contenu (CDN) nécessitent une préparation et une allocation de ressources appropriées. Avant de commencer les tests systématiques, assurez-vous de disposer des ressources et des outils nécessaires.

Ressources requises :

- Tester la distribution du CDN configurée pour refléter les paramètres de production

- Testez MediaTailor la configuration avec du contenu et des sources publicitaires connus
- Testez le contenu avec des caractéristiques prévisibles (durée, format, marqueurs de rupture publicitaire)
- Tester le serveur de décision publicitaire ou simuler des réponses ADS
- Plusieurs appareils de test et types de joueurs

Outils de test :

- `curl` pour tester les requêtes HTTP
- `ffprobe` pour la validation du manifeste HLS
- `mp4box` pour la validation du manifeste DASH
- Outils de développement de navigateurs pour l'analyse du réseau
- Lecteurs vidéo à end-to-end tester

Méthodologie de test systématique pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor les tests d'intégration des réseaux de diffusion de contenu (CDN) doivent suivre une approche systématique et progressive afin de garantir une couverture complète. Suivez cette approche structurée pour tester de manière exhaustive votre réseau de diffusion de contenu et votre MediaTailor intégration. Chaque phase s'appuie sur la précédente pour isoler les problèmes potentiels.

Pour des conseils supplémentaires sur les approches de test systématiques, voir [Testing for reliability in the AWS Well-Architected Framework](#).

Phase 1 : tester la MediaTailor connectivité directe

Commencez par tester les MediaTailor fonctionnalités sans impliquer le CDN pour établir une base de référence.

1. Testez les demandes de manifeste directement sur les MediaTailor points de terminaison :
 - Testez les demandes de playlist multivariantes HLS : `curl -v "https://your-emt-endpoint.mediatailor.region.amazonaws.com/v1/master/hls/config-name/master.m3u8"`

- Testez les requêtes DASH MPD : `curl -v "https://your-emt-endpoint.mediatailor.region.amazonaws.com/v1/dash/config-name/manifest.mpd"`
 - Vérifiez que les réponses au manifeste contiennent les marqueurs de rupture de publicité attendus
 - Vérifiez ce URLs point de segment pour corriger les origines
2. Vérifiez que l'insertion d'annonces fonctionne correctement :
- Testez avec différents paramètres de ciblage publicitaire
 - Vérifiez que les segments publicitaires sont correctement transcodés et disponibles
 - Vérifiez l'heure et la durée des pauses publicitaires
 - Testez le comportement de secours lorsque les publicités ne sont pas disponibles
3. Mesurez les performances de référence :
- Enregistrer les temps de réponse aux demandes de manifeste
 - Mesurez les temps de réponse des serveurs de décision publicitaire
 - Création et gestion de sessions de test

Critères de réussite : toutes les MediaTailor demandes directes renvoient des réponses HTTP 200 avec des manifestes correctement formatés contenant le contenu publicitaire attendu.

Phase 2 : tester l'intégration de base du CDN

Ajoutez un CDN au chemin de demande et testez les fonctionnalités de base.

1. Testez les demandes de manifeste via le CDN :
- Configurer le CDN avec des règles de routage simples
 - Testez les demandes de manifeste via les points de terminaison du CDN
 - Vérifiez que le CDN transmet correctement les demandes à MediaTailor
 - Vérifiez que les réponses au manifeste ne sont pas mises en cache (TTL = 0)
2. Routage des segments de test :
- Vérifier que les segments de contenu sont acheminés vers le serveur d'origine
 - Vérifier le routage des segments publicitaires vers le stockage MediaTailor publicitaire
 - Tester le comportement de mise en cache des segments
3. Comparez les performances du CDN aux performances directes :

- Mesurez les différences de temps de réponse
- Vérifiez les éventuelles différences de contenu dans les réponses
- Vérifiez que la gestion des erreurs fonctionne correctement

Critères de réussite : le CDN doit transmettre par proxy les requêtes aux serveurs d'origine MediaTailor et aux serveurs d'origine avec un impact minimal sur les performances.

Phase 3 : transfert des paramètres de requête de test

Ajoutez le transfert des paramètres de requête et testez la personnalisation des publicités.

1. Configurez le transfert des paramètres de requête sur le CDN :
 - Activez le transfert de tous les paramètres de requête vers MediaTailor
 - Initialisation de la session de test (l'identifiant de session est automatiquement généré MediaTailor lors de la première demande)
 - Testez avec des paramètres de ciblage personnalisés
2. Testez la personnalisation des publicités :
 - Vérifiez les différents paramètres et produisez des réponses publicitaires différentes
 - Codage des paramètres de test et caractères spéciaux
 - Vérifiez que les paramètres sont correctement transmis à ADS
3. Validez la gestion des sessions :
 - Création et persistance de sessions de test
 - Vérifier la cohérence des identifiants de session entre les demandes
 - Gestion de l'expiration des sessions de test

Critères de réussite : le contenu des annonces varie en fonction des paramètres de requête, et les sessions sont gérées correctement.

Phase 4 : Tester le transfert des en-têtes

Ajoutez le transfert d'en-tête dans le CDN et testez le ciblage spécifique à l'appareil.

1. Configurez le transfert d'en-têtes pour tous les en-têtes. Pour les exigences minimales, voir [En-têtes requis pour l'intégration du MediaTailor CDN](#).
2. Tester le ciblage des appareils :

- Testez avec différentes chaînes d'agent utilisateur (mobile, ordinateur de bureau, TV)
 - Vérifiez les réponses publicitaires spécifiques à l'appareil
 - Testez le ciblage géographique avec différentes adresses IP
3. Validez la gestion du CORS :
- Testez les en-têtes CORS pour vérifier leur compatibilité avec les lecteurs Web
 - Vérifiez que les demandes OPTIONS avant le vol fonctionnent correctement
 - Tester à partir de différents domaines

Critères de réussite : le ciblage par appareil et par zone géographique doit fonctionner correctement, et les lecteurs Web ne doivent pas rencontrer d'erreurs CORS.

Phase 5 : Test de scénario complet

Effectuez des tests sur plusieurs scénarios pour garantir un fonctionnement robuste.

1. Testez avec différents types de joueurs :
- Lecteurs HLS.js dans les navigateurs Web
 - Lecteurs Video.js compatibles avec HLS et DASH
 - Joueurs natifs sur appareils mobiles
 - Lecteurs Smart TV et décodeurs
2. Testez sur différents appareils et plateformes :
- Appareils mobiles (iOS, Android)
 - Navigateurs de bureau (Chrome, Firefox, Safari, Edge)
 - Appareils intelligents TVs et appareils de streaming
 - Différentes versions du système d'exploitation
3. Testez différents types de contenu :
- Contenu diffusé en direct
 - Contenu de vidéo à la demande (VOD)
 - Différents formats vidéo et débits
 - Contenu présentant différents types de pauses publicitaires
4. Testez des scénarios de ciblage publicitaire :
- Différents paramètres de ciblage démographique

- Ciblage géographique dans différentes régions
- Ciblage basé sur le temps (différents moments de la journée)
- Paramètres de logique métier personnalisés

Critères de réussite : toutes les combinaisons de joueurs et d'appareils doivent fonctionner correctement avec un ciblage publicitaire approprié.

Phase 6 : tests de charge et de performance

Validez les performances dans des conditions de charge réalistes.

Important

Avant de tester la charge, contactez [AWS Support](#) : avant d'effectuer des tests de charge et de performance, créez un ticket de AWS support pour informer l'équipe de MediaTailor service de vos tests prévus. Cela garantit :

- Le service est préparé pour les niveaux de charge attendus
- La capacité appropriée est disponible pendant votre fenêtre de test
- Vos tests ne seront pas confondus avec un incident de production
- Vous recevez des conseils sur les meilleures pratiques et les limites des tests

Incluez dans votre ticket d'assistance : les utilisateurs simultanés attendus, la durée du test, les régions géographiques et tout scénario spécifique que vous envisagez de tester.

1. Testez des scénarios d'utilisateurs simultanés :

- Simulez plusieurs spectateurs simultanés
- Testez le dimensionnement du CDN et les performances du cache
- Surveillez les performances du serveur d'origine en cas de charge
- Vérifiez que MediaTailor peut gérer des sessions simultanées

2. Mesurez les indicateurs de performance :

- Surveillez les temps de réponse sous charge
- Vérifiez que les taux d'accès au cache répondent aux attentes (> 80 % pour les contenus populaires)

- Mesurez le délai jusqu'à la première image pour différents scénarios
- Suivez les taux d'erreur pendant les pics de charge

3. Testez les scénarios de basculement :

- Tester le comportement lorsque ADS n'est pas disponible
- Tester le basculement du serveur Origin
- Vérifier la gestion des erreurs et la restauration
- Tester le basculement de l'emplacement périphérique du CDN

Critères de réussite : le système doit maintenir des performances acceptables sous la charge attendue, avec une dégradation progressive en cas de panne. Assurez-vous de contacter [AWS Support pour](#) qu'il approuve votre plan de test de charge avant son exécution.

Liste de contrôle des tests préalables au déploiement pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor L'intégration du réseau de diffusion de contenu (CDN) doit passer des tests complets avant le déploiement en production. Utilisez cette liste de contrôle avant de déployer des modifications de configuration en production.

Fonctionnalité de base :

- ☐ Les demandes de manifeste renvoient des réponses HTTP 200
- ☐ Les segments de contenu se chargent correctement
- ☐ Les segments publicitaires se chargent correctement
- ☐ Les pauses publicitaires apparaissent aux heures prévues
- ☐ La lecture passe en douceur entre le contenu et les publicités

Validation de la configuration :

- ☐ Les paramètres de requête sont transmis correctement
- ☐ Les en-têtes requis sont transmis correctement
- ☐ La mise en cache du manifeste est désactivée (TTL = 0)
- ☐ La mise en cache des segments est configurée de manière appropriée
- ☐ Les en-têtes CORS sont configurés pour les lecteurs Web

Tests multiplateformes :

- ☐ Testé sur des appareils mobiles
- ☐ Testé sur les navigateurs de bureau
- ☐ Testé avec différents types de joueurs
- ☐ Formats HLS et DASH testés

Validation des performances :

- ☐ Ticket d' AWS assistance créé pour l'approbation des tests de charge
- ☐ Les temps de réponse répondent aux objectifs de performance
- ☐ Les taux de réussite du cache sont acceptables
- ☐ Les taux d'erreur sont dans les limites acceptables
- ☐ La surveillance et les alertes sont configurées

Référence des outils et utilitaires de test pour le CDN et MediaTailor les intégrations

AWS Elemental MediaTailor les tests d'intégration du réseau de diffusion de contenu (CDN) nécessitent divers outils pour une validation et un débogage complets. Guide de référence pour les outils couramment utilisés dans les réseaux de diffusion de contenu et les tests MediaTailor d'intégration.

curl- Test des requêtes HTTP

Tester les demandes de manifeste : `curl -v "https://your-cdn-domain.com/v1/master/hls/config/master.m3u8"`

Test avec des en-têtes : `curl -H "User-Agent: TestAgent/1.0" "https://your-cdn-domain.com/..."`

Test avec paramètres : `curl "https://your-cdn-domain.com/...?aws.sessionId=test123"`

ffprobe- Validation du manifeste HLS

Validez la syntaxe HLS : `ffprobe -v quiet -print_format json -show_format "https://your-cdn-domain.com/master.m3u8"`

Vérifiez les informations du segment : `ffprobe -v quiet -show_entries packet=pts_time "segment.ts"`

mp4box- Validation du manifeste DASH

Validez le DASH MPD : `mp4box -info "https://your-cdn-domain.com/manifest.mpd"`

Vérifiez le chronométrage des segments : `mp4box -info segment.m4s`

Outils de développement de navigateurs

Surveillez les demandes réseau dans l'onglet Réseau

Vérifiez les erreurs CORS dans l'onglet Console

Inspecter les request/response en-têtes

Analyser les indicateurs de chronométrage et de performance

Ressources supplémentaires :

- [Résolution des problèmes CloudFront](#) - Guide complet de résolution des problèmes liés aux CDN
- [Augmenter CloudFront le taux de réussite du cache](#) - Conseils d'optimisation des performances
- [Surveiller AWS les ressources](#) - Meilleures pratiques en matière de surveillance des performances

Résoudre les problèmes liés à l'intégration MediaTailor du CDN

Ce guide de dépannage complet couvre les problèmes courants d'intégration du réseau de diffusion de contenu (CDN) dans toutes les AWS Elemental MediaTailor implémentations, notamment l'insertion de publicités côté serveur (SSAI), l'assemblage des canaux et l'intégration. AWS Elemental MediaPackage Lorsque votre CDN et votre MediaTailor intégration rencontrent des problèmes, utilisez cette approche diagnostique systématique pour identifier rapidement les causes profondes et mettre en œuvre des solutions validées.

Ce guide s'applique à toutes les intégrations de MediaTailor CDN, quel que soit votre flux de travail spécifique. Pour les problèmes spécifiques à des services ou à des flux de travail particuliers, consultez les sections de résolution des problèmes connexes référencées à la fin de ce guide.

Avant de commencer : Préparez les éléments suivants pour un dépannage efficace :

- Exemple de lecture URLs illustrant le problème
- Journaux d'accès au CDN datant de la période au cours de laquelle les problèmes sont survenus
- MediaTailor nom de configuration et AWS région
- Type et version du lecteur (par exemple, HLS.js 1.4.0, Video.js 8.0)
- Informations sur l'appareil et le navigateur à l'origine des problèmes

Sujets connexes :

- Pour la configuration opérationnelle et la préparation au dépannage, voir [Résoudre les problèmes liés à l'intégration du CDN](#)
- Pour l'analyse du journal et la référence aux codes d'erreur, voir [Référence d'analyse du journal d'intégration du CDN](#)
- Pour l'escalade et l'obtention d'une aide supplémentaire, voir [Bénéficiez d'une assistance pour l'intégration du CDN](#)

Liste de contrôle diagnostique pour les MediaTailor intégrations au CDN

AWS Elemental MediaTailor les problèmes d'intégration du réseau de diffusion de contenu (CDN) peuvent se manifester de différentes manières. Utilisez cette liste de contrôle pour identifier rapidement le type de problème que vous rencontrez :

1. Le problème concerne-t-il tous les spectateurs ou certains spectateurs en particulier ?
 - Tous les spectateurs → Probable problème de CDN ou MediaTailor de configuration
 - Nombre de spectateurs spécifiques → Probable problème de personnalisation ou de ciblage
2. Les manifestes se chargent-ils correctement ?
 - Non → Problème de routage ou de MediaTailor connectivité du CDN
 - Oui, mais contenu incorrect → Problème de mise en cache ou de personnalisation
3. Les segments se chargent-ils correctement ?
 - Échec des segments de contenu → Problème de connectivité avec Origin
 - Échec des segments publicitaires → Problème de diffusion ou de transcodage des annonces
4. Les annonces sont-elles insérées correctement ?
 - Aucune publicité n'apparaît → Vérifiez la connectivité et la configuration ADS

- De mauvaises annonces apparaissent → Vérifiez les paramètres de ciblage et de personnalisation des annonces
- Les publicités ne sont pas diffusées → Vérifiez le transcodage des publicités et la disponibilité des segments

5. La lecture est-elle fluide et ininterrompue ?

- Problèmes de mise en mémoire tampon → Vérifiez les performances du cache CDN et les temps de réponse d'origine
- Erreurs de lecture → Vérifier la syntaxe du manifeste et la disponibilité des segments
- Problèmes de transition publicitaire → Vérifiez le calendrier des pauses publicitaires et l'alignement des segments

6. Existe-t-il des codes ou des messages d'erreur spécifiques ?

- Erreurs HTTP 4xx → Vérifier le routage et la configuration du CDN
- Erreurs HTTP 5xx → Vérifier l'état du serveur d'origine et du MediaTailor service
- Erreurs spécifiques au joueur → Vérifier le format du manifeste et la compatibilité du lecteur

Prochaines étapes en fonction de votre diagnostic :

Problèmes de configuration du CDN

Pour un dépannage détaillé du routage et de la mise en cache du CDN, consultez [Résoudre les problèmes liés à l'intégration MediaTailor du CDN](#)

Problèmes de manifeste et de lecture

Pour la validation du manifeste et le dépannage de la lecture, voir [Procédures de test d'intégration CDN](#).

Problèmes d'insertion et de ciblage des publicités

Pour la résolution des problèmes spécifiques aux publicités, notamment la connectivité ADS et la diffusion des publicités, consultez la documentation de résolution des problèmes spécifiques à votre flux de travail.

Problèmes de performance et de surveillance

Pour l'analyse des performances et la configuration de la surveillance, voir [Surveillez les opérations et les performances du MediaTailor CDN](#).

Analyse du journal et codes d'erreur

Pour une analyse détaillée du journal et une référence aux codes d'erreur, consultez [Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor](#).

Tests et validation

Pour des procédures de test complètes, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#).

Si vous avez besoin d'une assistance immédiate ou si vous ne parvenez pas à résoudre le problème à l'aide des ressources liées, consultez [Bénéficiez d'une assistance et d'une aide au dépannage pour le CDN et les intégrations MediaTailor](#) les procédures d'escalade.

Procédures de test d'intégration CDN

Des tests appropriés sont essentiels avant de déployer votre intégration MediaTailor CDN en production. Ces procédures de test permettent d'identifier les problèmes de configuration, les problèmes de performances et les problèmes de compatibilité entre les différents appareils et plateformes.

Validation d'intégration de base

Effectuez ces tests fondamentaux pour vérifier que votre intégration CDN fonctionne correctement :

1. Livraison du manifeste de test :

- Demandez un manifeste via votre CDN et vérifiez qu'il renvoie une réponse valide
- Vérifiez que le manifeste contient le contenu attendu et les points d'insertion des annonces
- Vérifiez que le manifeste URLs utilise votre domaine CDN, pas l'origine
- Valider la syntaxe du manifeste à l'aide des outils de validation HLS ou DASH

2. Vérifiez la réécriture de l'URL :

- Vérifiez que le segment de contenu URLs des manifestes pointe vers votre domaine CDN
- Vérifiez que le segment publicitaire URLs pointe vers votre domaine CDN
- Assurez-vous que tous les membres de URLs votre famille sont correctement résolus

3. Testez la lecture du contenu :

- Diffusez du contenu via un lecteur vidéo et vérifiez la fluidité de la lecture
- Vérifiez que le contenu et les publicités sont diffusés sans interruption

- Vérifiez que les transitions entre le contenu et les publicités sont correctes
- Fonctionnalité de recherche de tests et de nettoyage

4. Validez le routage du CDN :

- Surveillez les journaux d'accès au CDN pour vous assurer que les demandes sont correctement acheminées
- Vérifiez que les hit/miss modèles de cache sont conformes aux attentes
- Vérifiez que les demandes d'origine ne se produisent que pour les erreurs de cache

Procédures de test avancées

Effectuez ces tests supplémentaires pour une validation complète :

1. Tests de compatibilité multiplateformes :

- Testez sur plusieurs appareils (ordinateur de bureau, mobile, tablette, téléviseur intelligent)
- Vérifiez la compatibilité entre les différents navigateurs
- Testez avec différents lecteurs vidéo (HLS.js, Video.js, lecteurs natifs)
- Valider sur différents systèmes d'exploitation

2. Tests de performance :

- Mesurer les temps de réponse aux demandes de manifeste (objectif : < 100 ms pour les demandes mises en cache)
- Testez les performances de chargement des segments sur différents débits
- Vérifiez que le temps de démarrage répond aux objectifs de performance
- Test dans différentes conditions de réseau

3. Validation du suivi des publicités :

- Vérifiez que les balises de suivi des publicités se déclenchent correctement
- Vérifiez l'exactitude des données d'analyse publicitaire
- Suivi des empreintes et de l'achèvement des tests
- Valider la fonctionnalité de clic

4. Test des conditions d'erreur :

- Tester le comportement lorsque l'origine est temporairement indisponible
- Vérifiez le traitement gracieux des demandes mal formées
- Testez les scénarios de basculement du CDN

- Valider la clarté et l'utilité des messages d'erreur

Création d'un environnement de test

Configurez un environnement de test qui reflète votre configuration de production pour une validation complète :

1. Configurez des distributions CDN distinctes pour les tests :

- Créez des distributions CDN de test avec les mêmes comportements de cache que ceux de production
- Configurez des origines de test qui reflètent votre configuration de production
- Utilisez des noms de domaine distincts pour éviter les conflits avec le trafic de production

2. Créez des MediaTailor configurations de test :

- Configurez des configurations de lecture de test avec les mêmes paramètres que ceux de production
- Configuration des points de terminaison du serveur de test et de décision
- Utilisez du contenu publicitaire de test qui correspond à vos formats d'annonces de production

3. Mettre en œuvre des processus de test systématiques :

- Création de listes de contrôle de test pour les modifications de configuration
- Documentez les procédures de test pour votre équipe
- Configurez des tests automatisés dans la mesure du possible

Testez sur plusieurs scénarios

Validez votre intégration dans différents scénarios et conditions pour garantir une couverture complète :

1. Testez avec plusieurs types de joueurs et appareils :

- Testez avec différents lecteurs vidéo (web, mobile, TV connectée)
- Validez sur différents systèmes d'exploitation et navigateurs
- Testez différentes conditions du réseau et vitesses de connexion

2. Créez des scripts de test automatisés :

- **Automatisez la validation des demandes de manifeste**

- Créez des scripts pour tester des scénarios d'insertion de publicités
 - Mettre en œuvre des tests de performance pour les scénarios à fort trafic
3. Validez le ciblage et la personnalisation des publicités :
- Testez avec différents profils d'utilisateurs et paramètres de ciblage
 - Valider l'intégration du serveur de décision publicitaire
 - Testez des scénarios de secours lorsque les publicités ne sont pas disponibles

Outils et techniques de test

Utilisez ces outils et techniques pour des tests efficaces :

Outils de développement de navigateurs

Utilisez l'onglet Réseau pour inspecter les requêtes et réponses HTTP

Surveillez la console pour détecter JavaScript les erreurs et les avertissements

Vérifier les en-têtes de réponse et le comportement de mise en cache

Vérifiez les informations temporelles pour l'analyse des performances

Tests en ligne de commande

Utilisez curl pour tester des en-têtes spécifiques URLs et inspecter les en-têtes :

```
curl -I "https://your-cdn-domain.com/path/to/manifest.m3u8"
```

Utilisez wget pour les tests de téléchargement et l'analyse du calendrier

Utilisez des outils tels que httpie pour des tests HTTP plus lisibles

Test du lecteur vidéo

Testez avec des implémentations à plusieurs joueurs

Utilisez les modes de débogage du joueur pour inspecter le comportement interne

Surveillez les événements liés aux joueurs et les rappels d'erreur

Valider le comportement de commutation adaptative à débit

Analyse et surveillance du CDN

- Surveillez les métriques du CDN en temps réel pendant les tests
- Consultez les journaux d'accès pour détecter les modèles de demandes
- Utilisez des outils de test spécifiques au CDN lorsqu'ils sont disponibles
- Configurez des alertes temporaires pour la validation des tests

Pour des méthodologies de test complètes supplémentaires et des approches de validation systématique, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#).

Résoudre les erreurs 404 du manifeste CDN pour MediaTailor

AWS Elemental MediaTailor Les erreurs 404 du manifeste du réseau de diffusion de contenu (CDN) constituent un problème d'intégration courant qui empêche le démarrage de la lecture. Cette section fournit un step-by-step dépannage en cas d'échec de livraison de manifestes.

Les demandes de playlist multivariante, de playlist multimédia ou de MPD renvoient 404 erreurs

Solution rapide (essayez d'abord) :

1. Vérifiez que le nom MediaTailor de configuration indiqué dans votre URL correspond exactement (en distinguant majuscules et minuscules)
2. Testez l'URL du manifeste directement MediaTailor sans CDN : `curl -v "https://your-
emt-endpoint.mediatailor.region.amazonaws.com/v1/master/hls/config-
name/master.m3u8"`
3. Si le test direct fonctionne, vérifiez les règles de routage du CDN pour les demandes de manifeste

Si la solution rapide ne fonctionne pas :

Symptômes : les joueurs ne démarrent pas la lecture, les demandes de manifeste renvoient des erreurs HTTP 404 dans les journaux du CDN.

Exemples de messages d'erreur :

- Console du navigateur : "Failed to load resource: the server responded with a status of 404 (Not Found)"
- Erreur du joueur : "MANIFEST_LOAD_ERROR" ou "NETWORK_ERROR"

- Journaux du CDN : GET /v1/master/hls/example-config/master.m3u8 404

Résolution :

Vérifiez que les règles de routage de votre CDN sont correctement configurées pour transférer les demandes de playlist multivariante, de playlist multimédia et de MPD à MediaTailor

Vérifiez que la MediaTailor configuration existe et qu'elle est correctement configurée.

Assurez-vous que les modèles de comportement de votre CDN correspondent aux chemins de demande de manifeste attendus (par exemple *.m3u8, *.mpd).

Diagnostiquez les problèmes et les erreurs de livraison de manifestes CDN pour MediaTailor

AWS Elemental MediaTailor les problèmes de diffusion du manifeste sur le réseau de diffusion de contenu (CDN) peuvent empêcher l'insertion et la lecture correctes des publicités. Si les spectateurs reçoivent des publicités incorrectes ou incohérentes dans les playlists multivariantes, les playlists multimédias, ou MPDs si :

1. Vérifiez les manifestes mis en cache :

- Vérifiez que les paramètres TTL sont définis sur 0 pour toutes les listes de lecture multivariantes, les listes de lecture multimédia et les chemins MPD
- Vérifiez que votre CDN ne met pas en cache les listes de lecture multivariantes, les listes de lecture multimédia ou MPDs malgré les paramètres TTL
- Vérifiez l'état du cache dans les journaux du CDN : les demandes de manifeste doivent s'afficher Miss ou RefreshHit non Hit

2. Vérifiez la configuration de routage du CDN :

- Confirmez que les demandes de manifeste sont acheminées vers les MediaTailor points de terminaison, et non mises en cache ou traitées depuis le point d'origine
- Vérifiez que les modèles de comportement du CDN correspondent correctement aux chemins du manifeste (*.m3u8, *.mpd)
- Vérifiez que les paramètres de requête sont transférés à des MediaTailor fins de personnalisation
- Testez le manifeste URLs directement MediaTailor pour isoler les problèmes liés au CDN et au service

3. Vérifiez la configuration du transfert d'en-têtes :

- Vérifiez que les en-têtes requis sont transmis (voir [En-têtes requis pour l'intégration du MediaTailor CDN](#))
- Vérifiez que l'en-tête de l'agent utilisateur est transféré pour un ciblage publicitaire spécifique à l'appareil
- Vérifiez que l' X-Forwarded-For-en-tête est transmis pour le ciblage géographique
- Assurez-vous que l'en-tête Accept-Encoding est transmis pour la prise en charge de la compression

4. Validez le contenu et la structure du manifeste :

- Vérifiez que les manifestes contiennent les marqueurs d'insertion de publicités attendus (EXT-X-CUE-OUT/IN pour HLS)
- Vérifiez que le segment URLs dans les manifestes utilise votre domaine CDN, pas les domaines d'origine
- Vérifiez que les segments publicitaires sont correctement insérés et accessibles
- Testez la syntaxe du manifeste à l'aide d'outils de validation (ffprobe pour HLS, mp4box pour DASH)

5. Testez différents scénarios :

- Testez avec une session différente IDs pour vérifier que la personnalisation fonctionne
- Effectuez des tests à partir de différents emplacements géographiques pour vérifier le ciblage géographique
- Testez avec différentes chaînes d'agent utilisateur pour vérifier le ciblage des appareils
- Comparez les réponses manifestes avec et sans CDN pour identifier les différences

Ressources de résolution des problèmes supplémentaires :

- Pour une configuration détaillée de la mise en cache du CDN, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
- Pour une configuration complète du routage CDN, voir [Configurer les comportements de routage du CDN pour MediaTailor](#)
- Pour les exigences relatives au transfert d'en-têtes, voir [En-têtes requis pour l'intégration du MediaTailor CDN](#)
- Pour l'analyse des journaux et les codes d'erreur, voir [Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor](#)

- Pour les procédures de test et de validation, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#)

Critères de réussite : une fois le problème résolu, les joueurs devraient démarrer la lecture normalement et les publicités devraient apparaître comme prévu. Les demandes de manifeste doivent renvoyer des codes d'état HTTP 200 dans les journaux du CDN, et les manifestes doivent contenir un contenu publicitaire correctement personnalisé.

Résoudre les problèmes de livraison et de chargement des segments CDN pour MediaTailor

AWS Elemental MediaTailor les problèmes de diffusion des segments du réseau de diffusion de contenu (CDN) peuvent entraîner une mise en mémoire tampon et des interruptions de lecture. Si les joueurs ne parviennent pas à charger des segments ou à mettre en mémoire tampon leur expérience :

1. Vérifiez les règles de routage du CDN :

- Vérifiez que les segments de contenu sont acheminés vers la bonne origine
- Vérifiez que les segments publicitaires sont acheminés vers le bon emplacement de stockage MediaTailor publicitaire
- Vérifiez que les extensions de fichiers de segment correspondent à vos modèles de comportement du CDN
- Vérifiez que le segment URLs des manifestes utilise le bon domaine CDN
- Pour obtenir des instructions détaillées sur la configuration des modèles de routage et de trajectoire comportementale, voir [Configurer les comportements de routage du CDN pour MediaTailor](#)

2. Vérifiez la configuration CORS :

- Pour les lecteurs Web, assurez-vous que votre CDN passe bien ou qu'il définit correctement les en-têtes CORS
- Testez à l'aide des outils de développement du navigateur pour identifier les erreurs liées au CORS
- Vérifiez que les demandes OPTIONS avant le vol sont traitées correctement

3. Accessibilité et performance des segments de test :

- Testez URLs directement chaque segment pour vérifier qu'il est accessible

- Vérifiez les temps de réponse des segments et identifiez les goulots d'étranglement en matière de performances
 - Vérifiez que la taille des fichiers de segment est adaptée à vos cibles de bande passante
 - Chargement des segments de test depuis différents emplacements géographiques
4. Validez le comportement de mise en cache du CDN pour les segments :
- Vérifiez que les segments de contenu disposent des paramètres TTL appropriés (généralement plus longs que les manifestes)
 - Vérifiez que les segments publicitaires sont mis en cache de manière appropriée en fonction des exigences de personnalisation
 - Surveillez les taux de réussite du cache pour le contenu et les segments publicitaires
 - Assurez-vous que les clés de cache n'incluent pas de paramètres inutiles qui réduisent l'efficacité du cache
5. Vérifiez la connectivité et l'état du serveur d'origine :
- Vérifiez que les serveurs d'origine répondent correctement aux demandes de segment
 - Vérifiez la capacité du serveur d'origine et les temps de réponse en cas de charge
 - Vérifiez que les serveurs d'origine disposent des fichiers de segment attendus
 - Testez les scénarios de basculement d'origine si plusieurs origines sont configurées
6. Résolvez les problèmes spécifiques aux segments publicitaires :
- Vérifiez que les segments publicitaires sont correctement transcodés et disponibles dans MediaTailor
 - Vérifiez que les segments publicitaires URLs sont correctement générés dans les manifestes
 - Testez le chargement des segments publicitaires à l'aide de différents paramètres de ciblage publicitaire
 - Surveillez les retards de transcodage des annonces susceptibles d'entraîner l'indisponibilité des segments
7. Validez la compatibilité et le comportement des joueurs :
- Testez le chargement des segments avec différents types de joueurs et différentes versions
 - Vérifiez les paramètres de la mémoire tampon du lecteur et les modèles de demandes de segments
 - Vérifiez la gestion des erreurs du lecteur en cas d'échec des demandes de segment
 - Tester la logique de commutation adaptative à débit et de sélection de segments

Ressources de résolution des problèmes supplémentaires :

- Pour la configuration du routage et du comportement du CDN, voir [Configurer les comportements de routage du CDN pour MediaTailor](#)
- Pour l'optimisation de la mise en cache du CDN, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
- Pour obtenir des conseils de configuration CORS, voir [Bonnes pratiques de sécurité de l'intégration CDN pour MediaTailor](#)
- Pour le suivi et l'analyse des performances, voir [Surveillez les opérations et les performances du MediaTailor CDN](#)
- Pour des procédures de test complètes, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#)
- Pour l'analyse des journaux et le diagnostic des erreurs, voir [Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor](#)

Critères de réussite : une fois résolus, les joueurs doivent charger les segments de manière fluide, sans interruption de la mise en mémoire tampon. Les demandes de segment doivent renvoyer des codes d'état HTTP 200 avec des temps de réponse appropriés, et le contenu et les segments publicitaires doivent être accessibles et correctement mis en cache.

Résoudre les problèmes de gestion et de suivi des sessions CDN pour MediaTailor

AWS Elemental MediaTailor la gestion des sessions du réseau de diffusion de contenu (CDN) est essentielle pour une personnalisation et un suivi appropriés des publicités. Si vous rencontrez des erreurs liées à la session ou si vous rencontrez un comportement incohérent entre les demandes :

1. Vérifiez la cohérence de l'identifiant de session :

- Vérifiez que votre lecteur conserve le même identifiant de session pour toutes les demandes relatives à une seule session de lecture
- Vérifiez les journaux du CDN pour confirmer que la session IDs est correctement transférée
- Assurez-vous que IDs les sessions sont correctement codées en URL dans les paramètres de requête
- Utilisez CloudWatch les journaux pour vérifier la cohérence des identifiants de session entre les demandes (voir les étapes de validation ci-dessous)

2. Validez l'initialisation de la session :

- Vérifiez que la première demande de manifeste crée correctement une session
- Vérifiez que le transfert des paramètres de session est correct (par exemple, `aws.sessionId`)
- Vérifiez l'initialisation de la session à l'aide des journaux de débogage (voir la configuration des journaux de débogage ci-dessous)

3. Activez la journalisation du débogage pour un dépannage détaillé des sessions :

- Pour les rapports côté serveur : ajoutez `?aws.logMode=DEBUG` à votre demande de lecture :

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.logMode=DEBUG
```

- Pour les rapports côté client : incluez `"logMode": "DEBUG"` dans le corps de votre demande d'initialisation de session
- Important : la `DEBUG` valeur distingue les majuscules et minuscules
- Maximum de 10 sessions de débogage actives autorisées simultanément

4. Utilisez CloudWatch les requêtes Logs pour valider le comportement des sessions :

- Vérifiez que la session de débogage est active :

```
fields @timestamp, @message  
| filter sessionId = "your-session-id-here"  
| filter eventType = "SESSION_INITIALIZED" # client-side reporting  
or mediaTailorPath like "/v1/master" # server-side reporting HLS  
or mediaTailorPath like "/v1/dash" # server-side reporting DASH
```

- Afficher tous les événements d'une session :

```
fields @timestamp, @message, eventType, mediaTailorPath  
| filter sessionId = "your-session-id-here"  
| sort @timestamp asc
```

- Vérifiez la génération du manifeste pour une session :

```
fields @timestamp, responseBody, @message  
| filter mediaTailorPath like "/v1/master/" and eventType = "GENERATED_MANIFEST"  
and sessionId = "your-session-id-here"
```

5. Tester le transfert des paramètres de session via le CDN :

- Testez les demandes de manifeste avec des paramètres de session directement par rapport MediaTailor (en contournant le CDN)
- Comparez le comportement des sessions avec et sans CDN pour identifier les problèmes de transfert
- Vérifiez que la configuration de transfert des paramètres de requête CDN inclut les paramètres liés à la session
- Vérifiez que le CDN ne met pas en cache les réponses qui devraient être spécifiques à la session

Messages d'erreur de session courants :

- `ConflictException(HTTP 409)` - Plusieurs demandes de playlist simultanées pour la même session. Solution : assurez-vous que votre joueur demande les playlists une par une conformément aux spécifications HLS
- `NotFoundException(HTTP 404)` - La session n'est pas disponible ou la configuration n'existe pas. Solution : Vérifiez la validité de votre configuration et réinitialisez la session
- `BadRequestException(HTTP 400)` - ID de session non valide ou demande mal formatée. Solution : vérifier le format des demandes et la validité de l'identifiant de session

Ressources de résolution des problèmes supplémentaires :

- Pour une configuration complète de la journalisation du débogage et une référence aux champs, voir [Génération AWS Elemental MediaTailor journaux de débogage](#)
- Pour CloudWatch des exemples de requêtes de journaux et d'analyse des journaux, voir [Ecriture AWS Elemental MediaTailor se connecte directement à Amazon CloudWatch Logs](#)
- Pour la configuration du transfert des paramètres de requête CDN, voir [Configurer les comportements de routage du CDN pour MediaTailor](#)
- Pour une référence complète des codes d'erreur, voir [Résolution des problèmes de lecture depuis MediaTailor](#)

Critères de réussite : une fois résolues, les sessions doivent s'initialiser correctement, maintenir une session IDs cohérente entre les demandes, et les journaux de débogage doivent indiquer les `SESSION_INITIALIZED` événements appropriés et la génération de manifestes sans erreur.

Résolvez les problèmes de synchronisation et de synchronisation des pauses publicitaires sur le CDN pour MediaTailor

AWS Elemental MediaTailor l'heure des pauses publicitaires du réseau de diffusion de contenu (CDN) doit être synchronisée avec précision avec les marqueurs de contenu. Si les annonces apparaissent à des heures incorrectes ou si l'heure des pauses publicitaires est incohérente :

1. Vérifiez les marqueurs de coupure de publicité dans le contenu :

- Vérifiez que les marqueurs SCTE-35 sont correctement placés dans votre contenu d'origine
- Vérifiez que la durée des interruptions publicitaires correspond à la longueur réelle du contenu publicitaire
- Vérifiez que la durée des pauses publicitaires correspond aux limites du contenu
- Validez le format du marqueur SCTE-35 et la précision du chronométrage dans vos manifestes d'origine
- Testez les marqueurs de coupure publicitaire avec différents types de contenu (en direct ou en VOD)

2. Vérifiez l'impact de la mise en cache du CDN sur le timing :

- Assurez-vous que le TTL du manifeste est défini sur 0 pour éviter toute dérive temporelle
- Vérifiez que les paramètres sensibles au temps ne sont pas mis en cache
- Vérifiez les problèmes de synchronisation des horloges entre la source de MediaTailor contenu et le CDN
- Surveillez la dérive temporelle dans les diffusions en direct de longue durée
- Vérifiez la synchronisation de l'heure du serveur CDN Edge avec NTP

3. Validez l'implémentation du marqueur SCTE-35 :

- Vérifiez que les EXT-X-DATERANGE étiquettes incluent les spécifications SCTE35 de SORTIE et de DURÉE appropriées
- Vérifiez la présence de marqueurs SCTE35 SCTE35 -OUT et -IN jumelés lorsque vous utilisez un chronométrage explicite
- Vérifiez que les horodatages de la date de début correspondent à l'heure réelle du contenu
- Testez différents formats de marqueurs SCTE-35 (marqueurs basés sur la durée ou marqueurs jumelés)

4. Testez le calendrier des pauses publicitaires dans différents scénarios :

- Comparez la durée des pauses publicitaires avec les MediaTailor demandes directes par rapport aux demandes CDN
 - Testez la cohérence des délais entre les différents emplacements périphériques du CDN
 - Vérifiez l'heure des pauses publicitaires en fonction des différents types de joueurs et de leurs comportements de mise en mémoire tampon
 - Surveillez la précision du chronométrage pendant les périodes de pointe
5. Déboguez les problèmes de synchronisation à l'aide des journaux et de la surveillance :
- Activez la journalisation du débogage pour suivre le temps de traitement des interruptions publicitaires
 - Surveillez CloudWatch les indicateurs relatifs au calendrier d'insertion des annonces
 - Vérifiez les journaux du CDN pour les modèles de demandes liés au calendrier
 - Utilisez les outils de débogage des joueurs pour vérifier l'heure des pauses publicitaires du point de vue du client

Tolérances temporelles attendues :

- L'heure des pauses publicitaires doit correspondre aux marqueurs SCTE-35 de votre contenu
- La durée de l'annonce doit correspondre à la durée spécifiée dans la réponse de votre serveur de décision publicitaire
- La synchronisation de l'horloge entre le contenu MediaTailor, la source et le CDN doit se faire dans un délai d'une seconde
- La synchronisation du marqueur SCTE-35 doit être précise à 100 ms près de la synchronisation réelle du contenu

Ressources de résolution des problèmes supplémentaires :

- Pour le format et la mise en œuvre du marqueur SCTE-35, voir [Intégrer une source de contenu pour l'insertion d' MediaTailor annonces](#)
- Pour la configuration de la journalisation du débogage et l'analyse du calendrier, voir [Génération AWS Elemental MediaTailor journaux de débogage](#)
- Pour la configuration de la mise en cache du CDN et son impact sur le timing, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

- Pour les procédures de test complètes, y compris la validation du calendrier, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#)
- Pour surveiller le temps d'insertion des publicités et les performances, voir [Surveillez les opérations et les performances du MediaTailor CDN](#)

Critères de succès : une fois résolus, les pauses publicitaires devraient apparaître exactement aux heures spécifiées par les marqueurs SCTE-35, avec un timing constant sur tous les sites périphériques du CDN et sur tous les types de joueurs. Les journaux de débogage doivent indiquer le moment précis du traitement des interruptions de publicité, sans dérive ni erreur de synchronisation.

Optimisez les performances du CDN et résolvez les problèmes de latence pour MediaTailor

AWS Elemental MediaTailor les performances d'intégration du réseau de diffusion de contenu (CDN) ont un impact direct sur l'expérience des spectateurs et la qualité de diffusion des publicités. Si vous constatez des temps de réponse lents ou une dégradation des performances :

Techniques de mesure du rendement

Avant de résoudre les problèmes de performance, établissez des mesures de référence et une surveillance continue :

1. Mesurez les indicateurs de performance clés :

- Temps de réponse : les demandes de manifeste doivent être traitées dans un délai de 200 ms, les demandes de segment dans un délai de 100 ms
- Taux d'accès au cache : segments de contenu > 95 %, segments publicitaires > 90 %
- Volume des demandes d'origine : doit être inférieur à 5 % du total des demandes lorsque le cache est optimisé
- Durée de réalisation de la première image : la lecture initiale devrait commencer dans les 2 à 3 secondes

2. Utilisez des outils de mesure du rendement :

- Tableaux de bord d'analyse du CDN : surveillez les performances du cache, les temps de réponse et les taux d'erreur
- CloudWatch métriques : suivez les métriques MediaTailor du service, y compris GetManifest .Latency

- Outils de développement de navigateurs : mesurez les performances côté client et la synchronisation du réseau
- Outils de ligne de commande : utilisez curl avec des options de chronométrage pour mesurer des demandes spécifiques

3. Mettre en œuvre une surveillance continue :

- Configurez des alertes de performance automatisées en cas de dégradation du temps de réponse
- Surveillez les performances dans différentes régions géographiques
- Suivez les performances pendant les périodes de pointe
- Comparez les indicateurs de performance avant et après les modifications de configuration

Ressources de mesure du rendement :

- Pour une configuration complète de surveillance des performances, voir [Surveillez les opérations et les performances du MediaTailor CDN](#)
- Pour les procédures de test de performance, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#)
- Pour CloudWatch les mesures et le suivi, voir [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#)

Problèmes de performance du cache CDN

Les problèmes de performance du cache font partie des problèmes d'intégration des CDN les plus courants. Ces problèmes affectent toutes les MediaTailor implémentations et peuvent avoir un impact significatif sur l'expérience utilisateur et les coûts.

Faible taux de réussite du cache

Symptômes : volume de demandes d'origine élevé, latence accrue, coûts de bande passante plus élevés, mauvaise expérience utilisateur

Valeurs cibles :

- Segments de contenu : taux d'accès au cache de 95 % ou plus
- Segments publicitaires : taux d'accès au cache de 90 % ou plus

- Manifestes : varie selon l'implémentation (les manifestes personnalisés ne doivent pas être mis en cache)

Causes courantes :

- Paramètres TTL incorrects pour différents types de contenu
- La configuration de la clé de cache inclut des paramètres de requête inutiles
- Les en-têtes de contrôle du cache depuis l'origine ne sont pas correctement configurés
- Invalidations ou purges fréquentes du cache
- Problèmes de distribution géographique (contenu non mis en cache aux emplacements périphériques)

Solutions :

1. Vérifiez et optimisez les paramètres TTL :

- Segments de contenu : définissez le TTL pour qu'il corresponde à la durée du segment ou plus
- Segments publicitaires : définissez le TTL sur 24 heures ou plus pour les publicités réutilisables
- Ressources statiques : définissez le TTL sur 24 heures ou plus

Pour des recommandations TTL complètes et des stratégies d'optimisation de la mise en cache, voir. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

2. Optimisez la configuration des clés de cache :

- Supprimer les paramètres de requête inutiles des clés de cache
- Assurez-vous que seuls les paramètres affectant le contenu sont inclus
- Normaliser l'ordre des paramètres et la distinction majuscules/

3. Vérifiez que les en-têtes de contrôle du cache d'origine sont correctement définis

4. Implémentez Origin Shield (ou une fonctionnalité CDN équivalente) pour les implémentations à fort trafic. La fonctionnalité Origin Shield est disponible dans les principales CDN versions, mais elle peut porter des noms différents (tels que CloudFront Origin Shield, Fastly Shield, Cloudflare Argo Tiered Cache). Si votre CDN ne propose pas cette fonctionnalité, vous pouvez l'activer MediaTailor lorsque vous contactez [AWS Support](#).

5. Passez en revue les stratégies d'invalidation du cache et réduisez les purges inutiles

Étapes de validation :

1. Surveillez les taux de réussite du cache à l'aide de tableaux de bord d'analyse du CDN
2. Test spécifique URLs avec curl pour vérifier les en-têtes du cache
3. Comparez le volume de demandes d'origine avant et après les modifications

Volume élevé de demandes d'origine

Symptômes : nombre étonnamment élevé de demandes atteignant le point MediaTailor d'origine, augmentation de la charge du serveur d'origine, augmentation des coûts

Schéma attendu : les demandes d'origine doivent représenter moins de 5 % du total des demandes des utilisateurs lorsque les taux d'accès au cache sont optimaux

Causes courantes :

- Erreurs de cache dues à de faibles valeurs TTL
- Fragmentation des clés de cache (trop de clés de cache uniques)
- Pics de trafic géographique dans les régions où le contenu n'est pas mis en cache
- Invalidations fréquentes du cache

Solutions :

1. Analyser les modèles de demandes pour identifier les causes des défaillances du cache
2. Optimisez les paramètres TTL en fonction du type de contenu et de la fréquence de mise à jour
3. Mettre en œuvre des stratégies de réchauffement du cache pour les nouveaux contenus
4. Pensez à l'implémentation d'Origin Shield (disponible dans CDNs les principales versions sous différents noms, voir [Implémentation d'Origin Shield](#) pour plus de détails)

Seuil d'alerte : définissez des alertes lorsque les demandes d'origine dépassent 10 % du total des demandes ou augmentent de 50 % par rapport à la base de référence

Résolution d'erreur HTTP courante

Les erreurs HTTP dans les intégrations CDN indiquent souvent des problèmes de configuration ou de service. Ces modèles d'erreur sont cohérents dans toutes les MediaTailor implémentations.

Erreur 404 Introuvable

Symptômes : les demandes de manifeste ou de segment renvoient le HTTP 404, les joueurs ne chargent pas le contenu, « MANIFEST_LOAD_ERROR » dans les journaux des joueurs

Causes courantes :

- Configuration d'origine du CDN incorrecte (URL de point de MediaTailor terminaison incorrecte)
- Modèles de chemin de comportement du cache manquants ou incorrects
- Problèmes de réécriture d'URL dans la configuration du CDN
- MediaTailor erreur de nom de configuration ou de point de terminaison de lecture
- Problèmes de synchronisation liés au contenu en direct (demande de segments futurs)

Étapes de diagnostic :

1. Testez la même URL directement par rapport à l'origine MediaTailor (contournez le CDN)
2. Vérifiez que la configuration d'origine du CDN correspond au point de terminaison MediaTailor de lecture
3. Vérifiez le comportement du cache CDN, les modèles de chemin et la priorité
4. Consultez les journaux d'accès au CDN pour connaître les détails du routage des demandes
5. Valider le nom de MediaTailor configuration et les paramètres régionaux

Solutions :

- Configuration d'origine du CDN correcte pour correspondre au point de terminaison MediaTailor de lecture
- Mettre à jour les modèles de chemin de comportement du cache pour acheminer correctement les demandes
- Corrigez les règles de réécriture d'URL, le cas échéant
- Vérifiez que MediaTailor la configuration existe et qu'elle est active

403 Erreurs interdites

Symptômes : les demandes renvoient le protocole HTTP 403, les messages d'accès sont refusés, l'authentification échoue

Causes courantes :

- Paramètres de requête manquants ou incorrects requis par MediaTailor
- Le CDN ne transmet pas les en-têtes ou les paramètres requis
- Restrictions relatives aux adresses IP ou blocage géographique
- Problèmes liés au jeton d'authentification (si vous utilisez un jeton signé URLs)

Solutions :

- Vérifiez que tous les paramètres de requête requis sont inclus et transmis
- Vérifiez la configuration du CDN pour le transfert d'en-têtes et de paramètres
- Passez en revue les restrictions IP et les paramètres géographiques
- Valider les jetons d'authentification et les processus de signature

400 erreurs liées à une mauvaise demande

Symptômes : les requêtes renvoient le HTTP 400, erreurs de requête mal formées, échecs de validation des paramètres

Causes courantes :

- Paramètres de requête mal formés ou problèmes de codage d'URL
- Valeurs de paramètres ou formats non valides
- Paramètres requis manquants pour des MediaTailor fonctionnalités spécifiques
- Limites de longueur d'URL dépassées

Solutions :

- Valider les formats des paramètres de requête et le codage des URL
- Vérifier les valeurs des paramètres par rapport aux exigences de MediaTailor l'API
- Assurez-vous que tous les paramètres requis sont inclus
- Vérifiez la longueur de l'URL et envisagez l'optimisation des paramètres

5xx Erreurs du serveur

Symptômes : les requêtes renvoient des erreurs HTTP 500, 502, 503 ou 504, des pannes de service intermittentes

Causes courantes :

- MediaTailor problèmes de service ou limites de capacité
- Problèmes de connectivité d'origine CDN
- Problèmes de délai d'attente dus à la lenteur des réponses d'origine
- Dégradation temporaire du service

Solutions :

- Consultez le AWS Service Health Dashboard pour connaître l'état du MediaTailor service
- Vérifiez la connectivité d'origine du CDN et les paramètres de délai d'expiration
- Implémenter une logique de nouvelle tentative avec un retard exponentiel
- Surveillez MediaTailor CloudWatch les indicateurs relatifs à l'état des services
- Contactez AWS le Support si les problèmes persistent

1. Mesurez les performances de référence :

- Testez les temps de réponse aux demandes de manifeste directement sur MediaTailor (cible : <200 ms)
- Mesurez les temps de réponse du CDN pour les demandes de manifeste (objectif : <100 ms pour les accès au cache)
- Vérifiez les temps de chargement des segments à la fois depuis l'origine et le CDN

2. Analysez les performances du CDN :

- Vérifiez le taux de réussite du cache pour les segments de contenu (objectif : > 80 % pour le contenu populaire)
- Vérifiez que le bouclier d'origine (ou une fonctionnalité CDN équivalente) est activé et configuré dans la même AWS région que votre origine. Différents CDNs utilisent des noms différents pour cette fonctionnalité
- Surveillez les performances de localisation périphérique et la distribution géographique du CDN

Critères de performance :

- Surveillez les temps de réponse à la génération des manifestes et comparez-les à vos performances de référence
- Les accès au cache du CDN sont nettement plus rapides que les requêtes d'origine
- Les temps de réponse de l'ADS ne devraient pas entraîner de retards dans la génération du manifeste

Ressources de résolution des problèmes supplémentaires :

- Pour des stratégies complètes d'optimisation des performances, voir [Guide d'optimisation des performances pour le CDN et MediaTailor les intégrations](#)
- Pour plus de détails sur l'implémentation d'Origin Shield, voir [Implémentation d'Origin Shield](#)

- Pour l'optimisation de la mise en cache du CDN, voir [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)
- Pour le suivi des performances et les indicateurs, voir [Surveillez les opérations et les performances du MediaTailor CDN](#)
- Pour les procédures de test de performance, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#)

Critères de réussite : une fois résolus, les temps de réponse doivent atteindre les objectifs de référence (manifestes inférieurs à 200 ms, segments inférieurs à 100 ms), les taux d'accès au cache doivent dépasser 90 % pour la plupart des types de contenu et le volume des demandes d'origine doit être inférieur à 5 % du total des demandes. Les performances doivent être cohérentes dans toutes les régions géographiques et tous les types d'appareils.

Corrigez le comportement incohérent du CDN sur les appareils et les plateformes pour MediaTailor

AWS Elemental MediaTailor L'intégration d'un réseau de diffusion de contenu (CDN) devrait permettre une diffusion uniforme des publicités sur tous les appareils et plateformes. Si les publicités se comportent différemment d'un appareil à l'autre :

1. Garantisiez un transfert d'en-tête cohérent dans tous les comportements du CDN.
 - Vérifiez que les en-têtes User-Agent et de ciblage personnalisé sont transmis de manière cohérente X-Forwarded-For
 - Vérifiez que les règles de transfert d'en-têtes s'appliquent à tous les comportements de cache pertinents
2. Vérifiez la compatibilité du lecteur avec la configuration de votre CDN.
 - Testez avec plusieurs types de joueurs (HLS.js, Video.js, joueurs natifs) pour identifier les problèmes spécifiques aux joueurs
 - Vérifiez les exigences d'en-tête spécifiques au joueur ou les différences de gestion des URL
3. Effectuez des tests avec plusieurs types d'appareils pour identifier les problèmes spécifiques à l'appareil.
 - Incluez les appareils mobiles, les tablettes, les navigateurs intelligents TV et les navigateurs de bureau dans vos tests
 - Testez différents systèmes d'exploitation et versions de navigateur
 - Vérifiez que le ciblage publicitaire spécifique à l'appareil fonctionne correctement

Si vous avez suivi ces étapes de résolution des problèmes et que vous avez toujours besoin d'aide, consultez [Bénéficiez d'une assistance pour l'intégration du CDN](#).

Préparation au dépannage

Configurez des outils et des processus pour simplifier le dépannage lorsque des problèmes d'intégration du CDN surviennent. La préparation proactive permet de résoudre les problèmes plus rapidement et plus efficacement en cas de problème.

Activez la journalisation complète

Des journaux détaillés sont essentiels pour diagnostiquer les problèmes d'intégration du CDN. Configurez la journalisation pour saisir les informations dont vous aurez besoin lors du dépannage.

1. Activez les journaux d'accès détaillés au CDN :

- Configurer la journalisation pour tous les comportements de cache qui traitent les MediaTailor demandes
- Inclure des chaînes de requête et des en-têtes personnalisés dans les entrées du journal
- Configurer des outils d'analyse des journaux pour identifier les modèles et les anomalies
- Activez les journaux en temps réel pour détecter immédiatement les problèmes lors d'événements en direct
- Configurer les politiques de conservation des journaux pour conserver l'historique des données de dépannage

2. Configurer la MediaTailor journalisation :

- Activez les journaux d'accès pour vos MediaTailor configurations
- Configuration de groupes de CloudWatch journaux pour une gestion centralisée des journaux
- Configurer des filtres de journal pour identifier les modèles d'erreur

3. Configurer la journalisation du serveur d'origine :

- Activez des journaux d'accès détaillés sur vos serveurs d'origine de contenu
- Inclure les en-têtes de demande et les codes de réponse dans les journaux
- Surveillez les indicateurs de performance du serveur d'origine

Ajouter des en-têtes de demande de diagnostic

Les en-têtes personnalisés permettent de suivre les demandes via votre CDN et d'identifier les problèmes de routage.

1. Configurez les en-têtes de diagnostic du CDN :

- Ajoutez un identifiant unique à chaque demande (par exemple, X-Request-ID)
- Inclure des informations spécifiques au CDN dans les en-têtes de demande
- Ajoutez des informations de localisation périphérique ou de point de présence (POP) pour suivre l'itinéraire géographique
- Inclure les en-têtes d'état du cache (Hit, Miss, RefreshHit) pour l'analyse du comportement du cache

2. Ajoutez des en-têtes de réponse pour le débogage :

- Inclure les en-têtes d'identification du serveur
- Ajoutez des informations de chronométrage pour l'analyse des performances
- Inclure les en-têtes de contrôle du cache pour les demandes de manifeste

Établir des indicateurs de performance de référence

Documentez les plages de performances normales pour identifier rapidement les anomalies lors du dépannage :

1. Enregistrez les mesures de base :

- Taux de réussite du cache pour différents types de contenu
- Percentiles de temps de réponse (P50, P95, P99)
- Taux d'erreur par code d'état
- Modèles de volume des demandes par heure de la journée

2. Documenter les attentes en matière de performance

- Taux de réussite du cache cible (95 % et plus pour le contenu, plus de 90 % pour les publicités)
- Temps de réponse acceptables (<100 ms en cache, <500 ms en origine)
- Taux d'erreur maximaux acceptables (< 1 % pour 4xx, < 0,1 % pour 5xx)

3. Créez des tableaux de bord de performance : configurez des tableaux de bord de surveillance qui présentent les mesures actuelles par rapport aux valeurs de référence.

Préparer des outils de résolution des problèmes

Configurez les outils et les autorisations d'accès nécessaires pour un dépannage efficace :

1. Outils de ligne de commande :

- `curl` pour tester les requêtes et réponses HTTP
- `dig` ou `nslookup` pour le dépannage du DNS
- Outils de validation HLS/DASH pour la vérification des manifestes
- Outils d'analyse de journaux (`grep`, `awk` ou analyseurs de journaux spécialisés)

2. Autorisations d'accès :

- Accès à la console de gestion du CDN pour la révision de la configuration
- MediaTailor accès à la console pour la vérification de la configuration
- CloudWatch accès aux métriques et à l'analyse des journaux
- Accès au serveur Origin pour le dépannage du backend

3. Documentation :

- Schémas d'architecture réseau
- CDN et documentation MediaTailor de configuration
- Informations de contact pour les procédures d'escalade
- Résolution des problèmes liés aux runbooks pour les scénarios courants

Guides de dépannage spécifiques au flux de travail

Ce guide de dépannage universel couvre les problèmes courants liés à toutes les intégrations de MediaTailor CDN. Pour les problèmes spécifiques à des flux de travail ou à des services particuliers, consultez ces ressources de dépannage spécialisées :

Résolution des problèmes liés à l'insertion de publicités côté serveur (SSAI)

Pour les problèmes spécifiques au SSAI, notamment les échecs d'insertion de publicités, les problèmes de transition des annonces et les problèmes de personnalisation, consultez la documentation de dépannage du SSAI spécifique au flux de travail.

Problèmes courants spécifiques à la SSAI :

- Échecs d'insertion d'annonces et pauses publicitaires vides
- Problèmes de synchronisation et de synchronisation liés à la transition des annonces
- Problèmes de personnalisation et de ciblage
- Divergences entre le suivi et les analyses des publicités

Résolution des problèmes d'assemblage des canaux

Pour les problèmes spécifiques à l'assemblage des canaux, notamment les problèmes de génération de manifestes et la fonctionnalité de décalage temporel, consultez la documentation du flux de travail d'assemblage des canaux.

Problèmes courants liés à l'assemblage des canaux :

- Erreurs de génération et de compilation de manifestes
- Problèmes liés à la fenêtre de décalage horaire et au fonctionnement du DVR
- Problèmes de disponibilité du contenu source et de basculement
- Planification du programme et synchronisation des métadonnées

MediaPackage résolution des problèmes d'intégration

Pour MediaPackage des problèmes spécifiques tels que le filtrage des manifestes et les problèmes liés aux terminaux EMP, consultez [Résolution des problèmes d'intégration au CDN](#).

Problèmes MediaPackage d'intégration courants :

- Erreurs de paramètres de filtrage du manifeste
- MediaPackage problèmes de connectivité des terminaux
- Problèmes de comportement du cache spécifiques à EMP
- MediaPackage problèmes d'authentification d'origine

CloudFront résolution de problèmes spécifiques

Pour les problèmes de configuration et d'installation CloudFront spécifiques, consultez la documentation CloudFront d'intégration.

CloudFront Problèmes courants :

- Configuration de la distribution et configuration du comportement du cache
- Identité d'accès à l'origine et configuration de sécurité
- CloudFront codes d'erreur et réponses spécifiques
- Restrictions géographiques et problèmes de localisation périphérique

Ressources supplémentaires :

- Pour obtenir des conseils sur l'optimisation des performances, voir [Optimisation des performances du CDN](#)

- Pour la configuration de la surveillance et des alertes, voir [Surveillance du CDN](#)
- Pour obtenir une assistance et une assistance générales, voir [Bénéficiez d'une assistance pour l'intégration du CDN](#)

Analyse du journal d'intégration du CDN et référence des codes d'erreur pour MediaTailor

AWS Elemental MediaTailor les journaux d'intégration du réseau de diffusion de contenu (CDN) fournissent des informations précieuses sur les performances et les erreurs. Ce guide couvre à la fois les journaux CDN (provenant de votre réseau de diffusion de contenu) et MediaTailor les journaux et codes d'erreur pertinents pour le dépannage de l'intégration du CDN. Utilisez cette référence lorsque vous devez comprendre ce que les journaux et les codes d'erreur de votre réseau de diffusion de contenu vous indiquent à propos de votre MediaTailor intégration. Ce guide vous aide à interpréter les entrées du journal et les messages d'erreur afin d'identifier la cause première des problèmes.

Sujets connexes :

- Pour les procédures de step-by-step dépannage, voir [Résoudre les problèmes liés à l'intégration du CDN](#)
- Pour des stratégies de surveillance et de prévention proactives, voir [Surveillez les opérations et les performances du MediaTailor CDN](#)
- Pour l'escalade et l'obtention d'une aide supplémentaire, voir [Bénéficiez d'une assistance pour l'intégration du CDN](#)
- Pour une configuration et des ressources de MediaTailor journalisation complètes, voir [MediaTailor ressources de configuration de journalisation](#)
- Pour une référence CloudFront sur le format du journal, voir [format du journal CloudFront d'accès](#)
- Pour une référence au code d'état [HTTP](#), voir [Codes d'état de réponse HTTP](#)

Techniques d'interprétation et d'analyse des logs CDN pour MediaTailor

AWS Elemental MediaTailor L'intégration d'un réseau de diffusion de contenu (CDN) génère des journaux détaillés qui vous aident à comprendre les modèles de demandes et à identifier les problèmes. Lorsque vous analysez les journaux du réseau de diffusion de contenu, concentrez-vous sur les indicateurs clés suivants :

Remarque : Les exemples suivants utilisent CloudFront la terminologie Amazon. D'autres fournisseurs de CDN peuvent utiliser des termes différents pour des concepts similaires.

Codes d'état HTTP

- 200- Demande réussie
- 404- Ressource introuvable (vérifiez les règles de routage et MediaTailor la configuration)
- 403- Accès refusé (vérifiez les paramètres de sécurité du CDN et les autorisations d'origine)
- 502/503/504- Erreurs du serveur d'origine (vérification de l'état MediaTailor du service et de la connectivité d'origine)

Indicateurs de comportement du cache

- Hit- Contenu diffusé depuis le cache du CDN
- Miss- Contenu extrait depuis l'origine
- RefreshHit- Contenu mis en cache validé avec Origin
- Error- La demande a entraîné une erreur

Modèles de demandes à surveiller

- Les demandes de manifeste doivent généralement résulter Miss ou être RefreshHit dues à de faibles paramètres TTL
- Les segments de contenu doivent s'afficher Hit pour les contenus populaires
- Des segments publicitaires peuvent s'afficher Miss en raison de la personnalisation

Surveillance de l'état des services

Lors de l'analyse des journaux du CDN et de la résolution des problèmes, vérifiez l'état du MediaTailor service pour déterminer si les problèmes sont liés au service :

AWS Tableau de bord du Service Health

Vérifiez l'état actuel du MediaTailor service et tout événement de service en cours

Accès : [AWS Service Health Dashboard](#)

À utiliser lorsque : vous constatez des erreurs 5xx ou des délais d'expiration de service généralisés dans les journaux du CDN

AWS Personal Health Dashboard

Afficher les notifications relatives à l'état du service et les événements de maintenance spécifiques au compte

Accès : [AWS Personal Health Dashboard](#)

À utiliser lorsque : vous avez besoin d'informations sur l'état du service ou de notifications de maintenance spécifiques au compte

Surveillance CloudWatch des métriques Amazon

Surveillez les indicateurs MediaTailor de santé des services à l'aide CloudWatch de mesures :

- `GetManifest.Errors`- Suivez les erreurs de génération de manifestes
- `GetManifest.Latency`- Surveillez les performances du temps de réponse
- `AdDecisionServer.Errors`- Surveillez les problèmes de connectivité des serveurs publicitaires
- `Origin.Errors`- Suivez les problèmes de connectivité du serveur d'origine

Accès : [CloudWatchconsole](#)

Utilisez-le lorsque : vous avez besoin de mesures de performance de service détaillées et de tendances historiques

Processus de résolution des problèmes liés à l'état du service

1. Consultez le AWS Service Health Dashboard pour connaître l'état actuel du service
2. Consultez le AWS Personal Health Dashboard pour les notifications spécifiques au compte
3. Surveillez CloudWatch les métriques pour les indicateurs de performance des services
4. Corrélisez l'état de santé du service avec les modèles de journalisation et les taux d'erreur du CDN
5. Si l'état du service est normal, concentrez-vous sur la configuration du CDN et les problèmes liés au serveur d'origine

Ressources d'analyse des journaux CDN

- [CloudFront journaux d'accès](#) - Guide complet sur le format et les champs des journaux CDN
- [CloudFront journaux en temps réel](#) - Streaming et analyse des journaux en temps réel

- [Analyse des données des CloudWatch journaux avec Logs Insights](#) - Requête et analyse avancées des journaux
- [Surveillez les ressources de charge de travail](#) - Modèles de surveillance Well-Architected

Rubriques

- [MediaTailor codes d'erreur et résolution des problèmes liés à l'intégration du CDN](#)
- [Outils d'analyse des journaux CDN et techniques de surveillance pour MediaTailor](#)
- [MediaTailor ressources de configuration de journalisation](#)

MediaTailor codes d'erreur et résolution des problèmes liés à l'intégration du CDN

AWS Elemental MediaTailor les codes d'erreur fournissent des informations spécifiques sur les problèmes d'intégration lorsqu'ils sont transmis via votre réseau de diffusion de contenu (CDN). Lorsque des erreurs sont MediaTailor renvoyées par le biais de votre réseau de diffusion de contenu, ces codes indiquent des problèmes spécifiques :

Remarque : Ces codes d'erreur sont renvoyés MediaTailor et transmis par votre CDN. Pour les erreurs spécifiques au CDN, consultez la documentation de votre fournisseur de CDN.

400 Requête erronée

Causes courantes : paramètres de requête mal formés, paramètres requis manquants, session non valide IDs

Vérification : configuration du transfert de chaînes de requêtes CDN, encodage des paramètres

403 Forbidden

Causes fréquentes : un joueur client demande un segment qui n'existe pas ou qui n'est pas autorisé à y accéder. Pour les segments publicitaires, MediaTailor il est possible que vous ayez spécifié un segment inexistant (contactez [AWS Support pour en](#) savoir plus). Pour les segments de contenu, les restrictions d'accès du fournisseur d'origine ou les problèmes d'authentification

Vérifiez : pour les segments publicitaires : contactez le [support AWS](#) si le segment généré n' MediaTailor est pas valide URLs. Pour les segments de contenu : vérifiez les autorisations du serveur d'origine, les informations d'authentification et les politiques d'accès. Vérifiez si les segments existent au niveau demandé URLs

404 – Non trouvé

Causes courantes : MediaTailor la configuration n'existe pas ou est inactive, chemin d'URL de lecture incorrect, manifeste ou demande de segment vers des ressources inexistantes

Vérification : vérifiez que MediaTailor la configuration existe et qu'elle est active, confirmez que l'URL de lecture correspond au formulaire `GetPlaybackConfiguration`, vérifiez les règles `ManifestEndpointPrefix` de routage du CDN, transférez les demandes aux points de terminaison appropriés MediaTailor

500 Erreur de serveur interne

Causes courantes : problèmes liés au serveur Origin, problèmes de connectivité ADS, erreurs de traitement des manifestes

Vérification : état du serveur Origin, validité de la réponse ADS, état du MediaTailor service

502 Passerelle défectueuse

Causes courantes : serveur d'origine inaccessible, réponse d'origine non valide

`UnsupportedManifestException` en raison de problèmes d'alignement des listes de lecture HLS (marqueurs SCTE non alignés sur les listes de lecture, marqueurs SCTE manquants sur certaines listes de lecture, temps de pause publicitaire incohérent selon les variantes de débit)

Vérification : règles de connectivité et de pare-feu au serveur Origin, résolution DNS, cohérence des playlists HLS sur toutes les variantes de débit, alignement des marqueurs SCTE-35 sur les mêmes segments dans toutes les playlists, vérification que toutes les playlists contiennent les marqueurs de coupure de publicité requis

Ressources d'analyse des codes d'erreur :

- [Résolution des problèmes MediaTailor](#) - Référence complète du code MediaTailor d'erreur
- [Résolution des problèmes de réponse aux CloudFront erreurs - Analyse des erreurs](#) spécifiques au CDN
- Codes [d'état des réponses HTTP - Référence complète des codes](#) d'état HTTP
- [Gestion des défaillances - Modèles de gestion](#) des erreurs Well-Architected

Outils d'analyse des journaux CDN et techniques de surveillance pour MediaTailor

AWS Elemental MediaTailor L'intégration d'un réseau de diffusion de contenu (CDN) génère d'importants volumes de données de journal qui nécessitent des outils et des techniques d'analyse efficaces. Utilisez ces approches pour analyser efficacement le réseau de diffusion de contenu et MediaTailor les journaux :

- Analyse par ligne de commande : utilisez des outils tels que `grepawk`, et `sort` pour filtrer et analyser les modèles de journalisation
- Amazon CloudWatch Logs Insights : CDN de requêtes et MediaTailor journaux avec une syntaxe similaire à celle de SQL pour une analyse avancée
- Outils tiers : pensez aux plateformes d'analyse des journaux pour une surveillance et des alertes complètes
- Tableaux de bord personnalisés : créez des visualisations qui combinent les métriques du CDN avec les données de performance MediaTailor

Si vous avez besoin d'une assistance supplémentaire pour l'analyse des logs ou l'interprétation de modèles d'erreur complexes, consultez [Bénéficiez d'une assistance pour l'intégration du CDN](#).

Outils et ressources d'analyse des journaux :

- [CloudWatch Logs Insights](#) - Requête de type SQL pour l'analyse des journaux
- [Amazon OpenSearch Service](#) - Recherche et analyse avancées dans les journaux
- [CloudWatch tableaux de bord](#) - Visualisation et surveillance personnalisées
- [Instrumentation des systèmes distribués pour une visibilité opérationnelle - Modèles](#) d'observabilité avancés
- Observabilité [de la charge de travail de conception - Guide d'observabilité](#) de Well-Architected

MediaTailor ressources de configuration de journalisation

Outre les journaux CDN, MediaTailor fournit des fonctionnalités de journalisation complètes pour surveiller l'insertion des publicités, la génération de manifestes et les interactions entre les services. Utilisez ces ressources pour configurer et analyser les MediaTailor journaux :

MediaTailor manifeste et journaux de service

Configurez et analysez les journaux pour la génération de manifestes, les interactions d'origine et les événements de service :

- [AWS Elemental MediaTailor description des journaux manifestes et types d'événements](#)- Référence complète pour les journaux de MediaTailor manifestes et les types d'événements
- [Visualisation AWS Elemental MediaTailor journaux](#)- Guide de visualisation et d'interprétation MediaTailor des journaux

Configuration des journaux vendus

Configurez la livraison flexible des journaux vers plusieurs destinations avec optimisation des coûts :

- [Utiliser des journaux vendus pour envoyer AWS Elemental MediaTailor journaux](#)- Configurer la livraison des journaux vers Amazon S3, Firehose ou Logs CloudWatch
- [Migrer votre AWS Elemental MediaTailor stratégie de journalisation](#)- Guide de migration pour les configurations de journalisation existantes

CloudWatch Intégration des journaux

Intégrez MediaTailor les journaux à CloudWatch des fins de surveillance et d'analyse :

- [Visualisation AWS Elemental MediaTailor journaux](#)- Configuration et analyse des CloudWatch journaux
- [CloudWatchInsights](#)- Interrogation et analyse avancées des journaux

Journalisation spécifique à la publicité

Surveillez les performances d'insertion des publicités et résolvez les problèmes liés aux publicités :

- Journaux d'interaction ADS - Surveillez les communications et les erreurs du serveur de décision publicitaire

Principales différences entre le CDN et les MediaTailor logs :

- Journaux du CDN : affiche request/response les modèles, le comportement du cache et les erreurs au niveau du réseau provenant de votre réseau de diffusion de contenu
- MediaTailor journaux : affiche les détails d'insertion des annonces, les événements de génération de manifestes, les interactions d'origine et les erreurs spécifiques au service

- Analyse combinée : utilisez les deux types de journaux ensemble pour une visibilité complète de votre flux de travail de streaming

Pour une surveillance complète qui combine à la fois le CDN et la MediaTailor journalisation, voir [Surveillez les opérations et les performances du MediaTailor CDN](#).

Automatisez MediaTailor et créez un CDN avec CloudFormation

L'automatisation à l' AWS Elemental MediaTailor aide d'un réseau de diffusion de contenu (CDN) AWS CloudFormation rationalise votre flux de travail d'insertion publicitaire et améliore la fiabilité.

Cette section explique comment utiliser AWS CloudFormation (AWS infrastructure en tant que service de code) pour configurer automatiquement un réseau AWS Elemental MediaTailor de diffusion de contenu (CDN). Bien que vous puissiez configurer manuellement cette intégration comme décrit dans les sections précédentes, son utilisation CloudFormation permet de gagner du temps et de réduire les erreurs en automatisant l'ensemble du processus à l'aide d'un seul modèle.

Si vous êtes nouveau dans ce CloudFormation domaine, il s'agit d'un service qui vous permet de créer un modèle de fichier qui définit toutes les ressources AWS dont vous avez besoin. Lorsque vous déployez ce modèle, il crée et configure CloudFormation automatiquement ces ressources pour vous, en veillant à ce qu'elles fonctionnent correctement ensemble.

Pour plus d'informations CloudFormation, consultez le [guide de CloudFormation l'utilisateur](#). Pour plus d'informations sur les types de MediaTailor ressources dans CloudFormation, voir [AWS: : MediaTailor Resource Type Reference](#).

Rubriques

- [Pourquoi utiliser CloudFormation for MediaTailor et intégrer un CDN](#)
- [Préparation au CloudFormation déploiement du CDN et MediaTailor des intégrations](#)
- [Déployer le CloudFormation modèle pour le CDN et les intégrations MediaTailor](#)
- [Utiliser les ressources CloudFormation déployées pour le CDN et l'intégration MediaTailor](#)
- [Testez et validez votre CloudFormation déploiement pour le CDN et l'intégration MediaTailor](#)
- [Résoudre les problèmes de CloudFormation déploiement courants pour le CDN et les intégrations MediaTailor](#)
- [Personnalisez le CloudFormation modèle pour le CDN et les intégrations MediaTailor](#)

- [AWS CloudFormation modèle de référence pour AWS Elemental MediaTailor l' CloudFront intégration à Amazon](#)

Pourquoi utiliser CloudFormation for MediaTailor et intégrer un CDN

AWS Elemental MediaTailor l'automatisation AWS CloudFormation offre des avantages significatifs aux professionnels de la diffusion qui gèrent les flux de travail de streaming. La configuration manuelle à l' MediaTailor aide d'un réseau de diffusion de contenu (CDN) peut être chronophage et source d'erreurs. Le recours à l' CloudFormation automatisation présente les avantages suivants.

- Cohérence : garantit que la même configuration est déployée à chaque fois, réduisant ainsi les erreurs humaines.
- Contrôle de version : stockez votre infrastructure sous forme de code dans un système de contrôle de version pour suivre les modifications.
- Déploiement rapide : déployez des configurations complexes en quelques minutes au lieu de plusieurs heures de configuration manuelle.
- Réplication de l'environnement : répliquez facilement les configurations dans les environnements de développement, de test et de production.
- Documentation : Le modèle lui-même sert de documentation de votre infrastructure.

Voici comment le flux de travail automatisé se compare à la configuration manuelle :

Configuration manuelle (plusieurs étapes)	Configuration automatisée (modèle unique)
Création d'une configuration de MediaTailor lecture	Déployer un CloudFormation modèle avec des paramètres
Création d'une CloudFront distribution	
Configuration des comportements du cache	
Configuration des configurations de sécurité	

Le flux de travail automatisé pour la configuration MediaTailor CloudFront suit les étapes suivantes :

1. Déployez le CloudFormation modèle avec l'origine de votre contenu et les paramètres de votre serveur publicitaire
2. CloudFormation crée et configure toutes les ressources nécessaires :
 - MediaTailor configuration de lecture pour l'insertion de publicités
 - CloudFront distribution avec des comportements de cache appropriés
 - Configurations de sécurité pour la protection du contenu
3. Utilisez les CloudFormation sorties pour accéder à votre flux publicitaire URLs
4. Diffusez votre contenu avec des publicités insérées dynamiquement

Préparation au CloudFormation déploiement du CDN et MediaTailor des intégrations

AWS Elemental MediaTailor déploiement AWS CloudFormation nécessitant des prérequis et des étapes de préparation spécifiques. Avant de commencer CloudFormation à travailler avec Amazon pour intégrer MediaTailor CloudFront, assurez-vous de disposer des éléments suivants.

- Un AWS compte avec des autorisations de création MediaTailor CloudFront, et CloudFormation des ressources
- Une origine de contenu sur laquelle votre contenu vidéo est hébergé (Amazon S3 ou un autre serveur d'origine, par exemple) AWS Elemental MediaPackage
- Un serveur de décision publicitaire (ADS) capable de répondre aux demandes VAST

Avant de déployer le CloudFormation modèle, collectez les paramètres obligatoires suivants :

AdServerUrl

URL du serveur publicitaire VAST pour l'insertion dynamique d'annonces. Un point de terminaison VAST statique est fourni pour les tests.

ContentOriginDomainName

Nom de domaine de l'origine de votre contenu sans protocole (par exemple *mediapackage-domain.mediapackagev2.us-west-2.amazonaws.com*, *mybucket.s3.amazonaws.com*, ou *custom-origin.example.com*). N'incluez pas de préfixes `http ://` ou `https ://` ni aucun chemin.

ContentOriginType

Type d'origine du contenu :

- `mediapackagev2` : Pour les origines AWS Elemental MediaPackage
- `s3` : Pour les origines des compartiments Amazon S3
- `personnalisé` : pour tout autre type d'origine

Le modèle créera plusieurs AWS ressources qui fonctionnent ensemble pour diffuser votre contenu sous forme de publicités personnalisées. Ce qui suit décrit le rôle de chaque composant :

Contrôle d'accès à l'origine

Le contrôle d'accès à l'origine (OAC) est une fonctionnalité de sécurité qui garantit que votre contenu n'est accessible que par le biais de votre serveur d'origine CloudFront, et non directement à partir de celui-ci. Cela permet de protéger votre contenu contre tout accès non autorisé.

Pour Amazon S3 Origins MediaPackage et Amazon S3, le modèle crée une ressource Origin Access Control (OAC) pour sécuriser l'accès à votre contenu.

MediaTailor configuration de lecture

La configuration de MediaTailor lecture est le composant principal qui gère l'insertion des publicités. Il reçoit du contenu provenant de votre origine, demande des publicités à votre serveur publicitaire et les combine dans un flux personnalisé pour chaque spectateur.

Le modèle crée une configuration de MediaTailor lecture avec les paramètres suivants :

- Source de contenu vidéo pointant vers votre CloudFront distribution
- URL du serveur de décision publicitaire configurée sur le point de terminaison VAST que vous avez spécifié
- Configuration de pré-diffusion en direct pour l'insertion de publicités pendant les diffusions en direct
- Configuration du CDN avec les préfixes d'URL de segment appropriés

CloudFront distribution

La CloudFront distribution diffuse votre contenu aux spectateurs du monde entier avec une faible latence. Il gère différents types de demandes (manifestes, segments de contenu, segments publicitaires) et les achemine vers les sources appropriées.

Pour les professionnels de la diffusion CDNs novices, voici quelques termes clés :

Origin

Un serveur sur lequel votre contenu original est stocké (comme MediaPackage Amazon S3)

Comportement de cache

Règles qui déterminent la manière dont les différents types de contenu sont mis en cache et diffusés

Politique de cache

Paramètres qui contrôlent la durée de mise en cache du contenu et les composants de demande qui affectent la mise en cache

Le modèle crée une CloudFront distribution avec les composants suivants :

- Trois origines :
 - Origine du contenu (MediaPackageAmazon S3 ou personnalisé)
 - MediaTailor origine des manifestes
 - MediaTailor origine des segments
- Comportements du cache avec des modèles appropriés :
 - Comportement par défaut pour les segments de contenu
 - Comportement des segments MediaTailor publicitaires (/tm/*)
 - Comportement pour les médias MediaTailor interstitiels (/v1/i-media/*)
 - Comportement pour les manifestes personnalisés (/v1/*)
 - Comportement des demandes de redirection de segment (/segment/*)
- Politiques de cache optimisées pour chaque comportement :
 - `CachingOptimized` pour du contenu pouvant être mis en cache
 - `CachingDisabled` pour des manifestes personnalisés
- Politiques de demande d'origine pour garantir le transfert correct des en-têtes
- Politiques d'en-tête de réponse pour le support CORS

Déployer le CloudFormation modèle pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor le déploiement à l'aide du AWS CloudFormation modèle est simple une fois que vous comprenez ce que le modèle va créer. Ce processus prend environ 15 à 30 minutes, la plupart du temps étant consacré à attendre le déploiement de la CloudFront distribution.

Pour déployer le CloudFormation modèle et configurer votre flux de travail d'insertion automatique d'annonces, procédez comme suit :

Pour déployer le MediaTailor CloudFormation modèle

1. Téléchargez le CloudFormation modèle depuis le AWS Elemental MediaTailor GitHub référentiel ou copiez-le depuis le [AWS CloudFormation modèle de référence pour AWS Elemental MediaTailor l' CloudFront intégration à Amazon](#).
2. Ouvrez la [CloudFormation console](#).
3. Choisissez Créer une pile > Avec de nouvelles ressources (standard).
4. Sous Spécifier le modèle, choisissez Télécharger un fichier modèle et chargez le modèle.
5. Entrez un nom de pile et fournissez des valeurs pour les paramètres requis :
 - AdServerUrl: URL de votre serveur publicitaire VAST (par exemple, `https://your-ad-server.com/vast`)
 - ContentOriginDomainName: nom de domaine de l'origine de votre contenu sans protocole (par exemple, `mediapackage-domain.mediapackagev2.us-west-2.amazonaws.com`)
 - ContentOriginType: Sélectionnez le type d'origine du contenu :
 - mediapackagev2 : Pour les origines AWS Elemental MediaPackage
 - s3 : Pour les origines des compartiments Amazon S3
 - personnalisé : pour tout autre type d'origine
6. Passez en revue la configuration et choisissez Create stack.
7. Attendez que la création de la pile soit terminée, ce qui prend généralement 5 à 10 minutes. Vous pouvez suivre les progrès dans l'onglet Événements.
8. Une fois terminé, accédez à l'onglet Sorties URLs pour trouver les manifestes HLS et DASH.

 Note

Si vous l'utilisez AWS Elemental MediaPackage comme origine de contenu, assurez-vous que vos MediaPackage points de terminaison sont correctement configurés et accessibles. Pour de plus amples informations, veuillez consulter [MediaPackage Intégration au CDN](#).

Utiliser les ressources CloudFormation déployées pour le CDN et l'intégration MediaTailor

AWS Elemental MediaTailor les ressources déployées par la AWS CloudFormation pile fournissent plusieurs résultats importants que vous utiliserez pour accéder à votre contenu grâce à l'insertion de publicités. Une fois la CloudFormation pile créée avec succès, vous devez comprendre comment utiliser les résultats pour accéder à votre contenu contenant des annonces insérées. Cette méthode est similaire à celle que vous utiliseriez MediaTailor URLs dans une configuration manuelle, mais le CloudFormation déploiement les fournit URLs automatiquement.

Une fois le déploiement réussi, la CloudFormation pile fournit plusieurs résultats importants que vous utiliserez pour accéder à votre contenu lors de l'insertion d'annonces :

CloudFrontDomainName

Le nom de domaine de votre CloudFront distribution (par exemple, *d1234abcdef.CloudFront.net*)

HlsManifestUrl

URL de base pour les manifestes HLS avec insertion d'annonces (par exemple, *https://d1234abcdef.CloudFront.net12345/v1/master////my-playback-config*)

DashManifestUrl

URL de base pour les manifestes DASH avec insertion d'annonces (par exemple, *https://d1234abcdef.CloudFront.net12345/v1/dash////my-playback-config*)

MediaTailorPlaybackConfigName

Nom de la configuration MediaTailor de lecture créée (par exemple *my-stack-PlaybackConfig*)

Construire une lecture URLs

Pour créer l'URL de diffusion complète de votre contenu contenant des publicités, vous devez associer l'URL de base des CloudFormation résultats à votre chemin de manifeste spécifique. Il s'agit d'une étape essentielle que les professionnels de la diffusion doivent comprendre, car elle connecte votre contenu existant au système d'insertion publicitaire.

1. Commencez par l'URL du manifeste approprié dans les sorties :

```
HlsManifestUrl: https://d1234abcdef.CloudFront.net/v1/master/12345/my-playback-  
config/
```

2. Ajoutez le chemin de votre manifeste spécifique :

```
Your manifest path: channel/index.m3u8
```

3. L'URL de lecture complète devient :

```
https://d1234abcdef.CloudFront.net/v1/master/12345/my-playback-config/channel/  
index.m3u8
```

Utilisez cette URL dans votre lecteur vidéo pour diffuser du contenu contenant des publicités insérées dynamiquement.

Tip

Si vous ne savez pas quel doit être le chemin de votre manifeste, vérifiez votre serveur d'origine. Pour les MediaPackage origines, il s'agit du chemin d'accès au manifeste HLS ou DASH de votre terminal. Pour les origines d'Amazon S3, il s'agit du chemin d'accès à votre fichier manifeste dans le compartiment.

Pour plus d'informations sur la structure des MediaTailor URL, consultez [Configurer l'intégration du CDN avec MediaTailor](#).

Configuration d'un lecteur vidéo

Une fois que vous avez obtenu votre URL de lecture, vous devez configurer un lecteur vidéo pour l'utiliser. Pour les professionnels de la diffusion, cela revient à configurer un lecteur pour n'importe

quel stream HLS ou DASH, mais le stream inclura désormais des publicités personnalisées. Voici un exemple simple d'utilisation du populaire HLS.js lecteur :

```
<!DOCTYPE html>
<html>
<head>
  <title>MediaTailor Playback Example</title>
  <script src="https://cdn.jsdelivr.net/npm/hls.js@latest"></script>
</head>
<body>
  <video id="video" controls style="width: 640px; height: 360px;"></video>

  <script>
    const video = document.getElementById('video');
    const mediaUrl = 'https://<replaceable>d1234abcdef.CloudFront.net</replaceable>/v1/master/<replaceable>12345</replaceable>/<replaceable>my-playback-config</replaceable>/<replaceable>channel/index.m3u8</replaceable>';

    if (Hls.isSupported()) {
      const hls = new Hls();
      hls.loadSource(mediaUrl);
      hls.attachMedia(video);
    } else if (video.canPlayType('application/vnd.apple.mpegurl')) {
      video.src = mediaUrl;
    }
  </script>
</body>
</html>
```

Vous pouvez également utiliser des lecteurs de diffusion professionnels tels que :

- Joueur JW
- Lecteur Bitmovin
- THEOplayer
- Video.js

Pour plus d'informations sur l'intégration des joueurs avec MediaTailor, consultez [MediaTailor exigences relatives à l'intégration des serveurs publicitaires](#).

Testez et validez votre CloudFormation déploiement pour le CDN et l'intégration MediaTailor

AWS Elemental MediaTailor la validation du déploiement est une étape essentielle pour les professionnels de la diffusion avant la mise en ligne. Cette section vous explique comment tester votre déploiement afin de vous assurer que les publicités sont correctement insérées et que le contenu est diffusé sans problème.

Après avoir déployé le CloudFormation modèle, procédez comme suit pour vérifier que votre configuration fonctionne correctement :

Pour tester votre CloudFront intégration MediaTailor et votre intégration

1. Vérifiez que toutes les ressources ont été créées correctement dans la CloudFormation console.
2. Vérifiez que la configuration de MediaTailor lecture est active dans la [MediaTailor console](#).
3. Vérifiez que la CloudFront distribution est déployée et activée dans la [CloudFront console](#).
4. Testez la lecture à l'aide d'un exemple de manifeste :
 1. Créez l'URL de lecture complète comme décrit dans [Construire une lecture URLs](#).
 2. Utilisez un lecteur vidéo compatible HLS ou DASH (comme VLC, JW Player ou le lecteur AWS console).
 3. Vérifiez que le contenu est diffusé et que les publicités sont insérées aux points de rupture attendus.
5. Vérifiez les connexions MediaTailor CloudWatch pour détecter d'éventuelles erreurs d'insertion d'annonces.

Lorsque vous testez l'insertion d'annonces, recherchez les indicateurs de succès suivants :

- Transitions fluides entre le contenu et les publicités
- Les publicités apparaissent aux points de rupture attendus (pré-lancement, mi-lancement, post-lancement)
- La qualité de l'annonce correspond à la qualité du contenu
- Aucune erreur de mise en mémoire tampon ou de lecture pendant les transitions publicitaires

Pour des procédures de test plus détaillées, voir [Comprendre le comportement d'insertion des AWS Elemental MediaTailor publicités](#). Pour des tests et une validation complets de l'intégration du CDN, voir [Tests et validation pour le CDN et MediaTailor les intégrations](#).

Résoudre les problèmes de CloudFormation déploiement courants pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor des problèmes de déploiement peuvent survenir même en cas d'automatisation lors du déploiement ou de la lecture. En tant que professionnel de la diffusion, comprendre comment résoudre ces problèmes vous aidera à maintenir un service de streaming fiable avec insertion de publicités.

Si vous rencontrez des problèmes lors de votre CloudFormation déploiement ou de l'CloudFront intégration qui en résulte MediaTailor , consultez les problèmes et solutions courants suivants :

CloudFormation problèmes de déploiement

La création de la pile échoue avec l'erreur « Échec de la création de ressources »

Causes possibles :

- Format de nom de domaine d'origine du contenu non valide
- Autorisations insuffisantes pour créer des ressources

Solution : Vérifiez l'erreur de ressource spécifique dans l'onglet CloudFormation Événements. Vérifiez que le nom de domaine d'origine du contenu est correctement formaté sans préfixes ni chemins de protocole. Assurez-vous que votre rôle IAM dispose des autorisations suffisantes pour créer toutes les ressources requises.

CloudFront le déploiement de la distribution prend beaucoup de temps

Cause : le déploiement complet des CloudFront distributions prend généralement 15 à 30 minutes.

Solution : il s'agit du comportement normal. Attendez que la distribution atteigne l'état « Déployé » avant de procéder au test.

Problèmes de lecture et d'insertion de publicités

Le contenu est diffusé mais aucune publicité n'est insérée

Causes possibles :

- Le serveur de décision publicitaire ne répond pas ou renvoie un VAST vide
- Le contenu ne contient pas de marqueurs publicitaires

Solution : vérifiez que votre serveur publicitaire est accessible et qu'il renvoie des réponses VAST valides. Vérifiez que votre contenu comporte des marqueurs publicitaires appropriés (marqueurs SCTE-35 pour le contenu en direct ou balises ad break pour la VOD).

403 Erreurs interdites lors de l'accès au contenu

Causes possibles :

- Le contrôle d'accès à l'origine n'est pas configuré correctement
- Problèmes liés au bucket d'origine ou aux autorisations des terminaux

Solution : pour les origines d'Amazon S3, vérifiez que la politique des compartiments autorise l'accès depuis la CloudFront distribution. Pour les MediaPackage origines, vérifiez que le contrôle d'accès aux origines est correctement configuré et que le point de terminaison est accessible.

Erreurs de lecture ou mise en mémoire tampon

Causes possibles :

- Les modèles de chemins de comportement du cache ne correspondent pas aux chemins de contenu
- Configuration du domaine d'origine incorrecte

Solution : vérifiez que vos comportements de cache présentent les modèles de chemin corrects pour acheminer les demandes vers les origines appropriées. Consultez CloudFront les journaux pour voir quelle origine traite les demandes et confirmez qu'il s'agit bien de celle attendue.

Pour les professionnels de la diffusion, ces conseils de dépannage supplémentaires peuvent être utiles :

- Utilisez Amazon CloudWatch Logs Insights pour rechercher des modèles d'erreur spécifiques dans MediaTailor les journaux

- Testez d'abord avec un simple serveur publicitaire VAST (comme celui fourni par défaut dans le modèle) avant d'utiliser votre serveur publicitaire de production
- Vérifiez les marqueurs publicitaires de votre contenu à l'aide de l'outil d'inspection des MediaTailor manifestes de la console
- Vérifiez le trafic réseau dans les outils de développement de votre navigateur pour vérifier si les demandes adressées au serveur publicitaire sont correctement envoyées

Pour un dépannage supplémentaire, consultez les CloudWatch journaux afin d'identifier les deux types d'erreurs MediaTailor et CloudFront d'identifier les erreurs spécifiques.

Personnalisez le CloudFormation modèle pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor la personnalisation du modèle permet aux professionnels de la diffusion d'adapter le AWS CloudFormation modèle aux exigences spécifiques du flux de travail. Bien que le modèle de base fonctionne pour de nombreux scénarios, ces personnalisations peuvent vous aider à répondre à des besoins plus complexes.

Les exemples ci-dessous montrent des extraits de code YAML que vous pouvez ajouter au modèle. Si vous ne connaissez pas le langage YAML ou CloudFormation la syntaxe, envisagez de faire appel à un développeur ou à un architecte de AWS solutions pour effectuer ces modifications.

Vous pouvez personnaliser le CloudFormation modèle pour répondre à vos exigences spécifiques en matière de flux de travail.

Ajouter ou modifier des origines

Pour les flux de diffusion qui utilisent plusieurs sources de contenu (telles que des sources principales et de sauvegarde, ou différentes bibliothèques de contenu), vous pouvez ajouter des origines supplémentaires à votre CloudFront distribution :

```
Origins:
  # Add a new origin for additional content
  - Id: SecondaryContentOrigin
    DomainName: secondary-content.example.com
    CustomOriginConfig:
      OriginProtocolPolicy: 'https-only'
      OriginSSLProtocols:
```

- TLSv1.2

Ajoutez ensuite un comportement de cache correspondant pour acheminer des modèles spécifiques vers cette origine :

```
CacheBehaviors:
  - PathPattern: '/secondary-content/*'
    TargetOriginId: SecondaryContentOrigin
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-CachingOptimized
```

Création de politiques de cache personnalisées

Pour les flux de diffusion soumis à des exigences de mise en cache spécifiques (comme les paramètres de sélection de qualité ou l'authentification du spectateur), vous pouvez créer des politiques de cache personnalisées au lieu d'utiliser des politiques de cache gérées. Pour obtenir des conseils détaillés sur les valeurs TTL et les stratégies de mise en cache, voir. [Optimisation de la mise en cache pour le CDN et les intégrations MediaTailor](#)

```
# Define a custom cache policy
CustomCachePolicy:
  Type: AWS::CloudFront::CachePolicy
  Properties:
    CachePolicyConfig:
      Name: !Sub '${AWS::StackName}-CustomCachePolicy'
      DefaultTTL: 86400 # 24 hours
      MaxTTL: 31536000 # 1 year
      MinTTL: 1 # 1 second
      ParametersInCacheKeyAndForwardedToOrigin:
        CookiesConfig:
          CookieBehavior: none
        HeadersConfig:
          HeaderBehavior: none
        QueryStringsConfig:
          QueryStringBehavior: whitelist
          QueryStrings:
            - quality
            - format

# Reference the custom policy in a cache behavior
CacheBehaviors:
```

```
- PathPattern: '/custom-path/*'
  TargetOriginId: ContentOrigin
  ViewerProtocolPolicy: 'https-only'
  CachePolicyId: !Ref CustomCachePolicy
```

Améliorez la MediaTailor configuration

Pour les flux de diffusion nécessitant des fonctionnalités avancées d'insertion de publicités, vous pouvez améliorer la MediaTailor configuration grâce à des options telles que le préchargement des publicités (pour réduire le temps de latence), les seuils de personnalisation et les publicités exceptionnelles.

```
MediaTailorPlaybackConfig:
  Type: AWS::MediaTailor::PlaybackConfiguration
  Properties:
    # Add ad prefetching for improved performance
    AvailSuppression:
      Mode: BEHIND_LIVE_EDGE
      Value: 00:00:00
    # Add personalization parameters
    PersonalizationThresholdSeconds: 2
    # Add bumper ads
    Bumper:
      StartUrl: https://example.com/bumper-start.mp4
      EndUrl: https://example.com/bumper-end.mp4
    # Other existing properties...
```

Pour plus d'informations sur les options MediaTailor de configuration, consultez [Utilisation AWS Elemental MediaTailor pour insérer des publicités](#).

Ajoutez des fonctionnalités de sécurité

Pour les flux de diffusion soumis à des exigences de sécurité spécifiques (comme les restrictions géographiques ou la protection contre les attaques DDoS), vous pouvez ajouter des AWS WAF intégrations et des restrictions géographiques :

```
# Create a AWS WAF Web ACL
WebACL:
  Type: AWS::WAFv2::WebACL
  Properties:
    Name: !Sub '${AWS::StackName}-WebACL'
```

```
Scope: CloudFront
DefaultAction:
  Allow: {}
VisibilityConfig:
  SampledRequestsEnabled: true
  CloudWatchMetricsEnabled: true
  MetricName: !Sub '${AWS::StackName}-WebACL'
Rules:
  - Name: RateLimitRule
    Priority: 0
    Action:
      Block: {}
    VisibilityConfig:
      SampledRequestsEnabled: true
      CloudWatchMetricsEnabled: true
      MetricName: RateLimitRule
    Statement:
      RateBasedStatement:
        Limit: 1000
        AggregateKeyType: IP

# Reference the AWS WAF Web ACL in the CloudFront distribution
CloudFrontDistribution:
  Type: AWS::CloudFront::Distribution
  Properties:
    DistributionConfig:
      WebACLId: !GetAtt WebACL.Arn
      # Add geo-restriction
      Restrictions:
        GeoRestriction:
          RestrictionType: whitelist
          Locations:
            - US
            - CA
            - GB
      # Other existing properties...
```

Pour plus d'informations sur les CloudFormation modèles, consultez le [guide de AWS CloudFormation l'utilisateur](#).

Pour obtenir des CloudFormation modèles et des exemples spécifiques à la diffusion, consultez le référentiel d'[outils GitHub AWS Media Services](#).

AWS CloudFormation modèle de référence pour AWS Elemental MediaTailor l' CloudFront intégration à Amazon

AWS Elemental MediaTailor l'intégration avec Amazon CloudFront peut être automatisée à l'aide du AWS CloudFormation modèle complet suivant :

```

AWSTemplateFormatVersion: '2010-09-09'
Description: | CloudFormation template that sets up AWS Elemental MediaTailor
  integration with CloudFront Distribution
  for server-side ad insertion. This template supports various content origins
  including MediaPackage, Amazon S3,
  and custom origins, making it versatile for different streaming architectures.

Parameters:
  AdServerUrl:
    Type: String
    Default: 'https://d1kbnkziz9rksx.CloudFront.net/VASTEndpoint.xml'
    Description: URL of the VAST ad server for dynamic ad insertion. Static VAST
    endpoint provided for testing.

  ContentOriginDomainName:
    Type: String
    Description: |
      Domain name of your content origin without protocol (e.g., mediapackage-
      domain.mediapackagev2.us-west-2.amazonaws.com,
      mybucket.s3.amazonaws.com, or custom-origin.example.com).
      Do not include http:// or https:// prefixes or any paths.
    AllowedPattern: "^(([a-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0-9\\-]*[a-zA-Z0-9])\\.)*([A-Za-
    z0-9]|[A-Za-z0-9][A-Za-z0-9\\-]*[A-Za-z0-9])$"
    ConstraintDescription: Must be a valid domain name (e.g., example.com) without
    protocol or path components. IP addresses are not allowed.

  ContentOriginType:
    Type: String
    AllowedValues:
      - mediapackagev2
      - s3
      - custom
    Default: mediapackagev2
    Description: |
      The type of content origin:
      - mediapackagev2: AWS Elemental MediaPackage V2
      - s3: Amazon S3 bucket

```

- custom: Any other custom origin

Resources:

```
#-----
# Origin Access Control (for securing MediaPackage V2 and Amazon S3 origins)
#-----
CloudFrontOriginAccessControl:
  Type: AWS::CloudFront::OriginAccessControl
  Condition: IsNotCustomOrigin
  Properties:
    OriginAccessControlConfig:
      Name: !Sub '${AWS::StackName}-OAC'
      OriginAccessControlOriginType: !Ref ContentOriginType
      SigningBehavior: always
      SigningProtocol: sigv4
      Description: Origin Access Control for content origin

#-----
# MediaTailor Playback Configuration
#-----
MediaTailorPlaybackConfig:
  Type: AWS::MediaTailor::PlaybackConfiguration
  Properties:
    Name: !Sub '${AWS::StackName}-PlaybackConfig'
    # The video content source should point to your CloudFront distribution
    VideoContentSourceUrl: !Sub 'https://${CloudFrontDistribution.DomainName}/'
    # The Ad Decision Server URL is where MediaTailor will request ads
    AdDecisionServerUrl: !Ref AdServerUrl
    # Configuration for pre-roll ads during live streams
    LivePreRollConfiguration:
      AdDecisionServerUrl: !Ref AdServerUrl
      MaxDurationSeconds: 30
    # CDN configuration for integrating with CloudFront
    CdnConfiguration:
      AdSegmentUrlPrefix: '/'
      ContentSegmentUrlPrefix: '/'
    # Set a reasonable manifest segment timeout
    ManifestProcessingRules:
      AdMarkerPassthrough:
        Enabled: false

#-----
# CloudFront Distribution
#-----
```

```

CloudFrontDistribution:
  Type: AWS::CloudFront::Distribution
  Properties:
    DistributionConfig:
      Enabled: true
      HttpVersion: http2and3
      IPV6Enabled: true
      Comment: !Sub 'Distribution for MediaTailor ad insertion with
${ContentOriginType} origin'

      # Default cache behavior points to the content origin
      DefaultCacheBehavior:
        TargetOriginId: ContentOrigin
        ViewerProtocolPolicy: 'https-only'
        # Using managed policies for optimal performance and simplicity
        CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
CachingOptimized
        OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
HostHeaderOnly
        ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
        Compress: true

      # Define all the origins needed for the workflow
      Origins:
        # Main content origin (MediaPackage, Amazon S3, or Custom)
        - Id: ContentOrigin
          DomainName: !Ref ContentOriginDomainName
          # Apply Origin Access Control for secure origins
          OriginAccessControlId: !If [IsNotCustomOrigin, !GetAtt
CloudFrontOriginAccessControl.Id, !Ref "AWS::NoValue"]
          # For custom origins, we need a CustomOriginConfig
          CustomOriginConfig:
            OriginProtocolPolicy: 'https-only'
            OriginSSLProtocols:
              - TLSv1.2
            OriginKeepaliveTimeout: 5
            OriginReadTimeout: 30
            HTTPPort: 80
            HTTPSPort: 443

        # MediaTailor Manifests Origin - handles manifest manipulation for ad
insertion
        - Id: MediaTailorManifests

```



```

    DomainName: !Sub 'manifests.mediataylor.${AWS::Region}.amazonaws.com'
    CustomOriginConfig:
      OriginProtocolPolicy: 'https-only'
      OriginSSLProtocols:
        - TLSv1.2
      OriginKeepaliveTimeout: 5
      OriginReadTimeout: 30
      # Origin Shield improves caching efficiency
    OriginShield:
      Enabled: true
      OriginShieldRegion: !Ref AWS::Region

# MediaTailor Segments Origin - handles personalized ads
- Id: MediaTailorSegments
  DomainName: !Sub 'segments.mediataylor.${AWS::Region}.amazonaws.com'
  CustomOriginConfig:
    OriginProtocolPolicy: 'https-only'
    OriginSSLProtocols:
      - TLSv1.2
    OriginKeepaliveTimeout: 5
    OriginReadTimeout: 30

# Cache behaviors to route specific request patterns to the right origin
CacheBehaviors:
  # Handle MediaTailor segment requests for ad content which are cache-able
  - PathPattern: '/tm/*'
    TargetOriginId: MediaTailorSegments
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
    CachingOptimized
    OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
    HostHeaderOnly
    ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
    CORS-with-preflight-and-SecurityHeadersPolicy
    Compress: true

  # Handle MediaTailor interstitial (SGAI) media requests which are cache-able
  - PathPattern: '/v1/i-media/*'
    TargetOriginId: MediaTailorManifests
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
    CachingOptimized
    OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
    HostHeaderOnly

```

```

        ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
        Compress: true

        # Handle MediaTailor Personalized manifests which are not cache-able
        - PathPattern: '/v1/*'
          TargetOriginId: MediaTailorManifests
          ViewerProtocolPolicy: 'https-only'
          CachePolicyId: 4135ea2d-6df8-44a3-9df3-4b5a84be39ad # Managed-
CachingDisabled
          OriginRequestPolicyId: 59781a5b-3903-41f3-afcb-af62929ccde1 # Managed-
AllViewer
          ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
          Compress: true

          # Handle MediaTailor segment *redirect* requests which are not cache-able
          (used for server side reporting)
          - PathPattern: '/segment/*'
            TargetOriginId: MediaTailorManifests
            ViewerProtocolPolicy: 'https-only'
            CachePolicyId: 4135ea2d-6df8-44a3-9df3-4b5a84be39ad # Managed-
CachingDisabled
            OriginRequestPolicyId: 59781a5b-3903-41f3-afcb-af62929ccde1 # Managed-
AllViewer
            ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
            Compress: true

Conditions:
  IsNotCustomOrigin: !Not [!Equals [!Ref ContentOriginType, 'custom']]

Outputs:
  CloudFrontDomainName:
    Description: Domain name of the CloudFront distribution
    Value: !GetAtt CloudFrontDistribution.DomainName

  HlsManifestUrl:
    Description: URL for HLS manifest with ads inserted (append your manifest path)
    Value: !Sub 'https://
${CloudFrontDistribution.DomainName}${MediaTailorPlaybackConfig.HlsConfiguration.ManifestEndpoint

  DashManifestUrl:
    Description: URL for DASH manifest with ads inserted (append your manifest path)

```

```
Value: !Sub 'https://  
${CloudFrontDistribution.DomainName}${MediaTailorPlaybackConfig.DashConfiguration.ManifestEndpointPath}  
  
MediaTailorPlaybackConfigName:  
Description: Name of the MediaTailor playback configuration  
Value: !Ref MediaTailorPlaybackConfig
```

Production-ready CloudFront configuration pour MediaTailor

Cette configuration CloudFront de distribution fournit tout ce dont vous avez besoin pour diffuser MediaTailor du contenu grâce à l'insertion de publicités côté serveur à grande échelle. Copiez cette configuration et personnalisez-la en fonction de vos origines et de vos besoins spécifiques.

Ce que cette configuration permet d'accomplir

Cette configuration crée une CloudFront distribution prête pour la production qui gère tous les types de MediaTailor demandes avec une mise en cache et des performances optimales. Il inclut trois origines (votre contenu, vos MediaTailor segments et vos MediaTailor manifestes) avec quatre comportements de cache qui acheminent correctement les demandes et mettent en cache le contenu de manière appropriée.

Quand utiliser cette configuration

Cette configuration est idéale pour le streaming en direct, la vidéo à la demande et les flux de travail hybrides qui nécessitent l'insertion de publicités côté serveur.

Three-origin architecture

MediaTailor utilise un modèle d'architecture à trois origines pour optimiser les performances de diffusion de contenu et d'insertion de publicités. Chaque origine répond à un objectif spécifique dans le flux de travail d'insertion d'annonces :

L'origine de votre contenu

C'est votre véritable source de contenu qui alimente MediaTailor. Par exemple, il peut s'agir de la AWS Elemental MediaPackage V2 ou d'un autre service de diffusion de contenu. Cette origine diffuse votre contenu original avant l'insertion de l'annonce. Parmi les exemples courants, citons :

- MediaPackage Configurations d'emballage V2
- Third-party réseaux de diffusion de contenu

- On-premises serveurs de streaming
- Compartiments Amazon S3 avec contenu statique

MediaTailor origine des segments

Cette origine utilise le nom d'hôte `segments.mediatailor.region.amazonaws.com` et diffuse les segments publicitaires eux-mêmes après les MediaTailor avoir encodés. Il s'agit des segments vidéo qui contiennent les publicités. Cette origine gère :

- Segments publicitaires transcodés dans le même format que votre contenu
- Demandes redirigées depuis le modèle de `/segment/*` chemin
- Segments publicitaires qui ont été traités pour l'insertion d'annonces côté serveur

MediaTailor origine des manifestes

Cette origine utilise le nom d'hôte `manifests.mediatailor.region.amazonaws.com` et peut être utilisée comme nom d'hôte régional pour les configurations de lecture dans la région spécifiée AWS . MediaTailor sélectionne la configuration de lecture correcte en fonction du chemin indiqué dans la demande. Cette origine fournit :

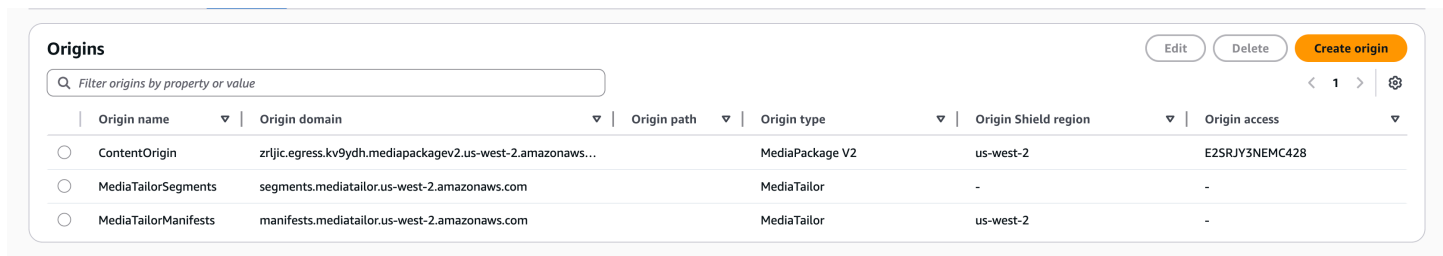
- Des manifestes HLS et DASH personnalisés avec insertion de publicités spécifiques à l'utilisateur
- Server-guided manifestes d'insertion de publicités (SGAI) pour le contenu pouvant être mis en cache
- Suivi des publicités et gestion des balises pour les rapports côté serveur

Avec le nom d'hôte d'origine `manifests.mediatailor.region.amazonaws.com`, vous pouvez avoir plusieurs configurations de lecture qui fonctionnent avec la même CloudFront distribution si elles se trouvent dans la région spécifiée et si vous incluez le nom de la configuration de lecture dans le chemin de la demande. Par exemple :

- `https://your-distribution.cloudfront.net/v1/master/playback-config-1/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/master/playback-config-2/manifest.m3u8`

Sélectionnez les politiques de demande d'origine en fonction du type de contenu afin d'éviter l'empoisonnement du cache tout en garantissant le bon fonctionnement. La principale distinction est entre le contenu pouvant être mis en cache et le contenu non cachable :

- Manifestes (non cachables) : AllViewer à utiliser pour transférer tous les en-têtes nécessaires au contenu dynamique. Comme les manifestes ne sont pas mis en cache, il n'y a aucun risque d'empoisonnement du cache.
- Segments (pouvant être mis en cache) : à utiliser None pour des performances optimales.
- Origines S3 : utilisation CORS-S3Origin pour les compartiments Amazon S3
- MediaPackage origines : Utilisation CORS-S3Origin pour les points de terminaison MediaPackage V2



The screenshot shows the 'Origins' management console in AWS MediaTailor. It includes a search bar, 'Edit', 'Delete', and 'Create origin' buttons, and a table with the following data:

	Origin name	Origin domain	Origin path	Origin type	Origin Shield region	Origin access
☐	ContentOrigin	zrljic.egress.kv9ydh.mediapackagev2.us-west-2.amazonaws.com		MediaPackage V2	us-west-2	E2SRJY3NEMC428
☐	MediaTailorSegments	segments.mediatailor.us-west-2.amazonaws.com		MediaTailor	-	-
☐	MediaTailorManifests	manifests.mediatailor.us-west-2.amazonaws.com		MediaTailor	us-west-2	-

Priorité et configuration du comportement du cache

MediaTailor nécessite des configurations de comportement de cache spécifiques pour gérer correctement les différents types de demandes. La priorité des comportements de cache est essentielle car les CDN les traitent dans l'ordre (du plus petit au plus grand) et utilisent le comportement pour le premier modèle de chemin correspondant. Il est essentiel de comprendre cette priorité pour résoudre les problèmes :

- Priorité 0 : la plupart des modèles spécifiques (similaires/tm/*) sont évalués en premier
- Numéros de priorité plus élevés : les modèles moins spécifiques sont évalués dans l'ordre
- Comportement par défaut : capture toutes les demandes qui ne correspondent pas à d'autres modèles

Si les demandes ne se comportent pas comme prévu, vérifiez que vos modèles de chemins ne se chevauchent pas de manière involontaire.

General	Security	Origins	Behaviors	Error pages	Invalidations	Tags	Logging
Behaviors Filter behaviors by property or value Save Move up Move down Edit Delete Create behavior							
	Preced...	Path patt...	Origin or origin group	Viewer pr...	Cache policy name	Origin request policy name	Response headers policy name
☐	0	/tm/*	MediaTailorSegments	HTTPS only	Managed-CachingOptimized	Managed-CORS-S3Origin	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	1	/i-media/*	MediaTailorManifests	HTTPS only	Managed-CachingOptimized	Managed-CORS-S3Origin	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	2	/v1/*	MediaTailorManifests	HTTPS only	Managed-CachingDisabled	Managed-AllViewerAndCloudFrontHeaders-2022-06	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	3	/segment/*	MediaTailorManifests	HTTPS only	Managed-CachingDisabled	Managed-AllViewerAndCloudFrontHeaders-2022-06	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	4	Default (*)	ContentOrigin	HTTPS only	Managed-CachingOptimized	CustomOriginRequestPolicyWithWithLLHLSDirectives	Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Priorité 0 : comportement du chemin des segments publicitaires

Ce comportement gère les demandes redirigées depuis le comportement du chemin du segment, en diffusant les segments publicitaires réels. CloudFront applique les comportements suivants à toutes les demandes comportant un modèle de `/tm/*` chemin. Il s'agit du comportement le plus prioritaire, car la diffusion de segments publicitaires est essentielle pour une diffusion ininterrompue.

- Schéma de trajectoire : `/tm/*`

Exemples d'URL correspondant à ce modèle :

- `https://your-distribution.cloudfront.net/tm/ad-segment-001.ts`
- `https://your-distribution.cloudfront.net/tm/transcoded-ad.m4s`
- Origine : origine que vous avez créée avec le `segments.mediatailor.region.amazonaws.com` domaine.

C'est MediaTailorSegments que montre l'exemple de la section précédente sur les origines.

- Politique de cache : `Managed-CachingOptimized`

Pour plus de détails sur ce qui est inclus dans la politique de cache CloudFront géré, consultez [CachingOptimized](#) le guide de CloudFront l'utilisateur. Vous pouvez également utiliser ces mêmes paramètres issus de la politique gérée de votre CDN tiers.

- Politique de demande d'origine : `None`
- Politique relative aux en-têtes de réponse : `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Pour plus de détails sur ce qui est inclus dans la politique relative aux en-têtes de réponse, consultez [CORS-with-preflight-and-SecurityHeadersPolicy](#) le guide de l' CloudFront utilisateur.

Adaptation de ces paramètres à d'autres CDN

Si vous utilisez un CDN autre que CloudFront, recherchez des paramètres équivalents permettant d'accomplir les tâches suivantes.

Correspondance des modèles de trajectoire

Configurer un comportement spécifique pour le modèle de `/tm/*` chemin afin de gérer les segments MediaTailor publicitaires

Configuration de la clé de cache

Incluez l'`Origin`-tête dans votre clé de cache pour vous assurer que les réponses sont mises en cache séparément pour différentes origines

Transfert d'en-tête

Transférer l'`Origin`-tête et les autres CORS-related en-têtes vers l'origine

Gestion des en-têtes de réponse

Configurez votre CDN pour vous assurer que l'`Access-Control-Allow-Origin`-tête est présent dans les réponses

La terminologie spécifique et les options de configuration varient selon le fournisseur de CDN, mais les principes sous-jacents restent les mêmes.

Priorité 1 : comportement d'insertion d' Server-guided annonces

Ce comportement gère les demandes [MediaTailor présentation et mise en œuvre de l'insertion d'annonces guidées par le serveur](#) (SGAI) lorsque les clients configurent le mode guidé, qui fournit des manifestes pouvant être mis en cache. CloudFront applique les comportements suivants à toutes les demandes comportant un modèle de `/v1/i-media/*` chemin. Le SGAi permet d'améliorer les performances de mise en cache car les manifestes ne sont pas spécifiques à l'utilisateur.

- Modèle de chemin : `/v1/i-media/*` (chemin iMedia pour SGAi)

Exemples d'URL correspondant à ce modèle :

- `https://your-distribution.cloudfront.net/v1/i-media/your-config/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/i-media/your-config/playlist.mpd`

- Origine : origine que vous avez créée avec le `manifests.mediatailor.region.amazonaws.com` domaine.

C'est MediaTailorManifests que montre l'exemple de la section précédente sur les origines.

- Politique de cache : `Managed-CachingOptimized`

Pour plus de détails sur ce qui est inclus dans la politique de cache CloudFront géré, consultez [CachingOptimized](#) le guide de CloudFront l'utilisateur. Vous pouvez également utiliser ces mêmes paramètres issus de la politique gérée de votre CDN tiers.

- Politique de demande d'origine : `None`
- Politique relative aux en-têtes de réponse : `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Pour plus de détails sur ce qui est inclus dans la politique relative aux en-têtes de réponse, consultez [CORS-with-preflight-and-SecurityHeadersPolicy](#) le guide de l' CloudFront utilisateur.

Priorité 2 : Comportement manifeste personnalisé

Ce comportement gère les demandes de manifeste personnalisées. CloudFront applique les comportements suivants à toutes les demandes comportant un modèle de `/v1/*` chemin. CloudFront applique les comportements suivants et ne met pas en cache les manifestes personnalisés car ils contiennent des URL de contenu publicitaire spécifiques aux spectateurs. Ces comportements s'appliquent à toutes les demandes comportant un modèle de `/v1/*` chemin. Il s'agit de la MediaTailor fonctionnalité de base qui permet à chaque spectateur de recevoir un manifeste unique avec une insertion publicitaire personnalisée.

- Modèle de chemin : `/v1/*` (MediaTailor requêtes V1 standard)

Exemples d'URL correspondant à ce modèle :

- `https://your-distribution.cloudfront.net/v1/master/your-config/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/dash/your-config/manifest.mpd`
- Origine : origine que vous avez créée avec le `manifests.mediatailor.region.amazonaws.com` domaine.

C'est MediaTailorManifests que montre l'exemple de la section précédente sur les origines.

- Politique de cache : `Managed-CachingDisabled`

Pour plus de détails sur ce qui est inclus dans la politique de cache, consultez [CachingDisabled](#) le guide de CloudFront l'utilisateur.

- Politique de demande d'origine : `AllViewer`

Pour les manifestes personnalisés, utilisez la `AllViewer` politique pour transférer tous les en-têtes nécessaires au contenu dynamique.

- Politique relative aux en-têtes de réponse : `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Pour plus de détails sur ce qui est inclus dans la politique relative aux en-têtes de réponse, consultez [CORS-with-preflight-and-SecurityHeadersPolicy](#) le guide de l' CloudFront utilisateur.

Accept-Encoding en-tête

Nous recommandons que votre CDN préserve l'Accept-Encoding en-tête des lecteurs. Cet en-tête donne des MediaTailor instructions sur la compression de manifestes personnalisés.

Dans CloudFront, la politique de demande `AllViewerAndCloudFrontHeaders` d'origine inclut le transfert de l'Accept-Encoding en-tête par le visualiseur. Si vous utilisez un autre CDN, assurez-vous qu'il préserve cet en-tête.

Voici comment MediaTailor gère l'Accept-Encoding en-tête.

- Appareils anciens : les anciens téléviseurs intelligents qui ne prennent pas en charge le format gzip n'envoient pas l' Accept-Encoding en-tête et MediaTailor renvoie donc des manifestes non compressés
- Appareils modernes : les iPhones, les navigateurs Chrome et autres clients modernes envoient l' Accept-Encoding en-tête, ce qui permet de MediaTailor compresser les manifestes avant la livraison

Priorité 3 : comportement de la trajectoire de la Server-side balise

Ce comportement gère les demandes MediaTailor qui entraînent des redirections pour [Server-side suivi](#). Ces demandes sont essentielles pour le suivi des balises. Chaque demande doit donc être traitée par MediaTailor. CloudFront applique les comportements suivants à toutes les demandes comportant un modèle de `/segment/*` chemin.

- Schéma de trajectoire : `/segment/*`

Exemples d'URL correspondant à ce modèle :

- `https://your-distribution.cloudfront.net/segment/tracking-beacon-123`
- `https://your-distribution.cloudfront.net/segment/ad-request-456.ts`
- Origine : origine que vous avez créée avec le `manifests.mediatailor.region.amazonaws.com` domaine.

C'est MediaTailorManifests ce qui montre l'exemple de la section précédente sur les origines.

- Politique de cache : `Managed-CachingDisabled`

Pour plus de détails sur ce qui est inclus dans la politique de cache, consultez [CachingDisabled](#) le guide de CloudFront l'utilisateur.

- Politique de demande d'origine : `AllViewer`

Pour les demandes de balises côté serveur, utilisez la `AllViewer` politique pour transférer tous les en-têtes nécessaires au suivi. Comme ces demandes ne sont pas mises en cache, il n'y a aucun risque d'empoisonnement du cache.

- Politique relative aux en-têtes de réponse : `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Pour plus de détails sur ce qui est inclus dans la politique relative aux en-têtes de réponse, consultez [CORS-with-preflight-and-SecurityHeadersPolicy](#) le guide de l' CloudFront utilisateur.

Lors MediaTailor du traitement de ces demandes, il renvoie une réponse de redirection 302 avec un chemin pointant vers l'emplacement réel du segment. Par exemple, une demande `/segment/ad123.ts` peut être redirigée vers `/tm/encoded-ad-segment.ts` l'origine du segment.

Priorité 4 : Comportement du chemin d'origine du contenu

Si le chemin de la demande ne correspond à aucun des autres modèles, CloudFront applique le comportement par défaut. Ce comportement envoie les demandes directement à l'origine du contenu, sans traitement depuis MediaTailor. Cela permet d'accéder directement à l'origine de votre contenu (comme la MediaPackage V2) en cas de besoin. CloudFront applique les comportements suivants à toutes les demandes qui n'incluent aucun des modèles de chemin précédents.

- Schéma de trajectoire : `(*)`

- Origine : origine que vous avez créée avec le domaine correspondant à l'origine de votre contenu.

C'est ContentOrigin que montre l'exemple de la section précédente sur les origines.

- Politique de cache : Managed-CachingOptimized

Pour plus de détails sur ce qui est inclus dans la politique de cache CloudFront géré, consultez [CachingOptimized](#) le guide de CloudFront l'utilisateur. Vous pouvez également utiliser ces mêmes paramètres issus de la politique gérée de votre CDN tiers.

Note

Pour les implémentations HLS à faible latence, envisagez d'utiliser une politique de mise en cache personnalisée avec des directives HLS à faible latence (LLH) au lieu de la politique standard. CachingOptimized

- Politique de demande d'origine : None
- Politique relative aux en-têtes de réponse : Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Bien que le comportement d'origine du contenu par défaut ne présente généralement pas les mêmes risques d'empoisonnement du cache CORS que le comportement des segments publicitaires, il est toujours recommandé d'appliquer la politique des en-têtes de Managed-CORS-with-preflight-and-SecurityHeadersPolicy réponse et d'inclure l'Origin-en-tête dans votre clé de cache. Cela garantit une gestion cohérente du CORS pour tous les types de contenu et évite les problèmes de lecture potentiels sur les lecteurs Web.

Pour les segments de contenu, la politique de Managed-CachingOptimized cache fournit de bonnes performances tandis que la politique des en-têtes de Managed-CORS-with-preflight-and-SecurityHeadersPolicy réponse garantit une gestion correcte du CORS. Cette combinaison permet une mise en cache efficace tout en maintenant la compatibilité avec les lecteurs Web qui nécessitent des en-têtes CORS.

L'application d'une gestion CORS cohérente à la fois aux segments publicitaires et aux segments de contenu permet de créer une expérience de lecture plus fiable et de simplifier le dépannage. Sans une configuration CORS appropriée, les joueurs peuvent rencontrer des comportements incohérents lors de la transition entre le contenu et les publicités.

Pour plus de détails sur ce qui est inclus dans la politique relative aux en-têtes de réponse, consultez [CORS-with-preflight-and-Security-HeadersPolicy](#) le guide de l' CloudFront utilisateur.

Bénéficiez d'une assistance et d'une aide au dépannage pour le CDN et les intégrations MediaTailor

AWS Elemental MediaTailor Les problèmes d'intégration du CDN peuvent être complexes à diagnostiquer et à résoudre. Utilisez ce guide lorsque vous avez besoin d'une aide supplémentaire pour résoudre des problèmes liés au CDN et à MediaTailor l'intégration que vous ne pouvez pas résoudre par le biais d'un dépannage en libre-service.

Avant de passer à l'escalade, essayez les options de libre-service suivantes :

- Suivez les étapes de résolution des problèmes décrites dans [Résoudre les problèmes liés à l'intégration du CDN](#)
- Analysez vos journaux et vos codes d'erreur à l'aide de [Référence d'analyse du journal d'intégration du CDN](#)
- Passez en revue votre configuration de surveillance avec [Configuration des outils de surveillance](#)
- Consultez le [guide de MediaTailor dépannage](#) pour les problèmes spécifiques au service
- Recherchez dans [AWS Re:Post](#) des problèmes similaires et des solutions communautaires
- Consultez le [centre de connaissances AWS](#) pour découvrir les modèles d'intégration courants

Rubriques

- [Quand signaler les problèmes liés au MediaTailor CDN à AWS Support](#)
- [Collectez des informations MediaTailor sur le CDN avant de contacter le support](#)
- [MediaTailor Ressources et canaux de soutien à l'intégration du CDN](#)
- [MediaTailor Meilleures pratiques en matière d'assistance à l'intégration du CDN](#)

Quand signaler les problèmes liés au MediaTailor CDN à AWS Support

AWS Elemental MediaTailor Les problèmes d'intégration du CDN doivent être signalés au AWS Support lorsque le dépannage en libre-service ne résout pas le problème. Envisagez de contacter le AWS Support lorsque :

Remarque : le support technique nécessite un plan de AWS support payant. Pour en savoir plus sur les plans de support et les temps de réponse, consultez les [plans de support AWS](#).

- Les problèmes affectent la production, le trafic ou les revenus
- Vous avez suivi toutes les étapes de résolution des problèmes sans résolution
- Les modèles d'erreur suggèrent des problèmes de niveau de service
- Vous avez besoin d'assistance pour des scénarios de configuration complexes
- Les problèmes de performances persistent malgré les efforts d'optimisation

Collectez des informations MediaTailor sur le CDN avant de contacter le support

AWS Elemental MediaTailor Les cas de support relatifs à l'intégration du CDN nécessitent des informations de configuration et de diagnostic spécifiques pour un dépannage efficace. Avant de contacter AWS le Support, collectez les informations suivantes pour accélérer la résolution :

Conseil : Le fait de disposer de ces informations avant de créer votre dossier de support réduira considérablement le temps de résolution et aidera les ingénieurs du support à comprendre votre configuration spécifique.

- MediaTailor détails de configuration :
 - Nom de la configuration et AWS région
 - ARN de configuration de lecture
 - URL ADS et type d'intégration
- Détails de configuration du CDN :
 - ID de distribution ou nom de configuration du CDN
 - Configurations du comportement du cache pour les manifestes et les segments
 - Configuration d'origine et règles de routage
- Informations sur les erreurs :
 - Messages d'erreur spécifiques et codes d'état HTTP
 - Horodatage en cas de problème
 - URLs Échantillon illustrant le problème
 - Entrées relatives au CDN et au MediaTailor journal liées au problème
- Informations sur les tests :

- Mesures que vous avez déjà prises pour résoudre le problème
- Appareils et lecteurs sur lesquels le problème se produit
- Si le problème concerne l'ensemble du contenu ou des flux spécifiques
- Fréquence et schéma du problème (intermittent, constant, basé sur le temps)

MediaTailor Ressources et canaux de soutien à l'intégration du CDN

AWS Elemental MediaTailor L'assistance à l'intégration du CDN est disponible via plusieurs canaux pour vous aider à résoudre les problèmes et à optimiser votre mise en œuvre :

- AWS Support : créez un dossier de support via la console AWS de gestion pour obtenir une assistance technique

Accès : [AWS Support Center](#)

- AWS Re:post : Plateforme de questions-réponses axée sur la communauté pour les questions connexes et le soutien de la AWS communauté

Accès : [AWS Re:post](#)

- AWS Documentation : guides complets MediaTailor et services CDN

Accès : [MediaTailordocumentation](#) et [CloudFrontdocumentation](#)

- AWS Formation : cours et certifications pour les services médias et l'optimisation des CDN

Accès : [AWS formation et certification](#)

- AWS Centre de connaissances : articles sélectionnés sur les AWS problèmes courants et les meilleures pratiques

Accès : [AWS Knowledge Center](#)

- AWS Trusted Advisor : recommandations automatisées pour l'optimisation et les meilleures pratiques

Accès : [AWS Trusted Advisor](#)

- AWS Personal Health Dashboard : notifications relatives à l'état du service et à la maintenance

Accès : [AWS Personal Health Dashboard](#)

MediaTailor Meilleures pratiques en matière d'assistance à l'intégration du CDN

AWS Elemental MediaTailor Les cas de support relatifs à l'intégration du CDN sont résolus plus efficacement lorsque vous suivez ces bonnes pratiques. Pour obtenir la résolution la plus rapide :

- Choisissez le niveau de gravité approprié en fonction de l'impact commercial
- Fournissez toutes les informations pertinentes lors de la soumission initiale de votre dossier
- Incluez des exemples spécifiques et des étapes de reproduction
- Joindre les fichiers journaux pertinents et les captures d'écran de configuration
- Répondre rapidement aux demandes d'informations supplémentaires des ingénieurs de support

Ressources d'assistance supplémentaires :

- [AWS Gestion des dossiers de support](#) - Guide de création et de gestion des dossiers de support
- [AWS Plans de support - Comparez les fonctionnalités des plans](#) de support et les temps de réponse
- [Préparez-vous à supporter les charges de travail - Recommandations](#) de Well-Architected pour la préparation opérationnelle
- [AWS Well-Architected](#) Framework - Meilleures pratiques pour créer et exploiter des charges de travail sur AWS

Sécurité dans AWS Elemental MediaTailor

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez de centres de données et d'architectures réseau conçus pour répondre aux exigences des organisations les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le [modèle de responsabilité partagée](#) décrit ceci 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 AWS les services dans le AWS Cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Third-party les auditeurs testent et vérifient régulièrement l'efficacité de notre sécurité dans le [AWS cadre des programmes de](#) . Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS Elemental MediaTailor, consultez la section [AWS Services inclus dans le champ d'application du programme de conformité](#) .
- **Sécurité dans le cloud** — Votre responsabilité est déterminée par le AWS service que vous utilisez. 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 MediaTailor. Les rubriques suivantes vous montrent comment procéder à la configuration MediaTailor pour atteindre vos objectifs de sécurité et de conformité. Vous apprenez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser vos MediaTailor ressources.

Rubriques

- [Protection des données dans AWS Elemental MediaTailor](#)
- [Gestion des identités et des accès pour AWS Elemental MediaTailor](#)
- [Validation de conformité pour AWS Elemental MediaTailor](#)
- [Résilience dans AWS Elemental MediaTailor](#)
- [Sécurité de l'infrastructure dans MediaTailor](#)
- [Cross-service prévention adjointe confuse](#)
- [Enregistrement et surveillance dans MediaTailor](#)

Protection des données dans AWS Elemental MediaTailor

Le [modèle de responsabilité partagée](#) AWS s'applique à la protection des données dans AWS Elemental MediaTailor. Comme décrit dans ce modèle, AWS est responsable de la protection de l'infrastructure mondiale qui gère tous les AWS Cloud. 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 plus d'informations sur la confidentialité des données, consultez la FAQ sur la confidentialité [des données et les conditions légales de la région](#) . Pour plus d'informations sur la protection des données en Europe, consultez le [Centre du règlement général sur la protection des données \(RGPD\)](#).

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou Gestion des identités et des accès AWS (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.
- SSL/TLS À utiliser pour communiquer avec AWS les ressources. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation des activités des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour enregistrer AWS des activités, consultez la section [Utilisation des CloudTrail sentiers](#) dans le Guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS cryptage, ainsi que tous les contrôles de sécurité par défaut qui s'y trouvent 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 par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez [Norme FIPS \(Federal Information Processing Standard\) 140-3](#).

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 Nom. Cela inclut lorsque vous travaillez avec MediaTailor ou Services AWS utilisez la console, l'API ou les kits de AWS développement logiciel (SDK). AWS CLI Toutes les données

que vous entrez 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.

Chiffrement des données

AWS Elemental MediaTailor ne chiffre ni ne déchiffre les données dans le cadre de la gestion des manifestes de contenu ou de ses communications avec les serveurs, les CDN ou les lecteurs. MediaTailor n'exige pas que vous fournissiez des données clients ou d'autres informations sensibles.

Ne saisissez pas d'informations sensibles, telles que les numéros de compte client, les informations de carte de crédit ou les informations de connexion, dans des champs libres ou des paramètres de requête. Cela s'applique à toutes les utilisations de AWS Elemental MediaTailor, y compris la console, l'API, les SDK et le AWS Command Line Interface (AWS CLI). Toutes les données que vous entrez dans le service peuvent être récupérées pour être insérées dans des journaux de diagnostic.

Lorsque vous fournissez une URL à un serveur externe, n'incluez pas les informations d'identification non chiffrées dans l'URL pour valider votre demande adressée au serveur.

Gestion des identités et des accès pour AWS Elemental MediaTailor

Gestion des identités et des accès AWS (IAM) permet à un Service AWS administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (possède des autorisations) à utiliser MediaTailor les ressources. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- [Public ciblé](#)
- [Authentification par des identités](#)
- [Gestion de l'accès à l'aide de politiques](#)
- [Comment ? AWS Elemental MediaTailor fonctionne avec IAM](#)

- [Identity-based exemples de politiques pour AWS Elemental MediaTailor](#)
- [Resource-based exemples de politiques pour AWS Elemental MediaTailor](#)
- [AWS politiques gérées pour AWS Elemental MediaTailor](#)
- [Utilisation de rôles liés à des services pour MediaTailor](#)
- [Résolution des problèmes AWS Elemental MediaTailor identité et accès](#)

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir [Résolution des problèmes AWS Elemental MediaTailor identité et accès](#))
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir [Comment ? AWS Elemental MediaTailor fonctionne avec IAM](#))
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir [Identity-based exemples de politiques pour AWS Elemental MediaTailor](#))

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identité. Vous devez être authentifié en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en tant qu'identité fédérée à l'aide d'informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), de l'authentification unique ou d'informations d'identification. Google/Facebook Pour plus d'informations sur la connexion, consultez [Connexion à votre Compte AWS](#) dans le Guide de l'utilisateur Connexion à AWS .

Pour l'accès par programmation, AWS fournit un SDK et une interface de ligne de commande pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez [Signature AWS Version 4 pour les demandes d'API](#) dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez avec une seule identité de connexion appelée utilisateur Compte AWS root qui dispose d'un accès complet à toutes Services AWS les ressources. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches qui requièrent des informations d'identification de l'utilisateur racine, consultez [Tâches qui requièrent les informations d'identification de l'utilisateur racine](#) dans le Guide de l'utilisateur IAM.

Identité fédérée

Il est recommandé d'obliger les utilisateurs humains à utiliser la fédération avec un fournisseur d'identité pour accéder à l' Services AWS aide d'informations d'identification temporaires.

Une identité fédérée est un utilisateur de votre annuaire d'entreprise, de votre fournisseur d'identité Web ou Directory Service qui accède à l' Services AWS aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées 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. Pour plus d'informations, 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 qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, voir [Exiger des utilisateurs humains qu'ils utilisent la fédération avec un fournisseur d'identité pour accéder à AWS l'aide d'informations d'identification temporaires](#) dans le Guide de l'utilisateur IAM.

[Les groupes IAM](#) spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez [Cas d'utilisation pour les utilisateurs IAM](#) dans le Guide de l'utilisateur IAM.

Rôles IAM

Un [rôle IAM](#) est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Vous pouvez assumer un rôle en [passant d'un rôle utilisateur à un rôle IAM \(console\)](#) ou en appelant une opération AWS CLI ou AWS API. Pour plus d'informations, consultez [Méthodes pour endosser un rôle](#) dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

Gestion de l'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 définit les autorisations lorsqu'elles sont associées à une identité ou à une ressource. AWS évalue ces politiques lorsqu'un directeur fait une demande. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations les documents de politique JSON, consultez [Vue d'ensemble des politiques JSON](#) dans le Guide de l'utilisateur IAM.

À l'aide de politiques, les administrateurs précisent qui a accès à quoi en définissant 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. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l'opération.

Identity-based politiques

Identity-based les politiques sont des documents de politique d'autorisations JSON que vous associez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez [Définition d'autorisations IAM personnalisées avec des politiques gérées par le client](#) dans le Guide de l'utilisateur IAM.

Identity-based les politiques peuvent être des politiques en ligne (intégrées directement dans une identité unique) ou des politiques gérées (politiques autonomes associées à plusieurs identités). Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez [Choix entre les politiques gérées et les politiques en ligne](#) dans le Guide de l'utilisateur IAM.

Resource-based politiques

Resource-based les politiques sont des documents de stratégie JSON que vous joignez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de

compartiment Amazon S3. 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. Vous devez [spécifier un principal](#) dans une politique basée sur les ressources.

Resource-based les politiques 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.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir les autorisations maximales accordées par les types de politiques les plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez [Limites d'autorisations pour des entités IAM](#) dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCP) : spécifient les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez [Politiques de contrôle de service](#) dans le Guide de l'utilisateur AWS Organizations .
- Politiques de contrôle des ressources (RCP) : définissent les autorisations maximales disponibles pour les ressources de votre organisation. Pour plus d'informations, consultez [Politiques de contrôle des ressources \(RCP\)](#) dans le Guide de l'utilisateur AWS Organizations .
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez [Politiques de session](#) 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 ? AWS Elemental MediaTailor fonctionne avec IAM

Avant d'utiliser IAM pour gérer l'accès à MediaTailor, découvrez quelles fonctionnalités IAM sont disponibles. MediaTailor

Fonctionnalités IAM que vous pouvez utiliser avec AWS Elemental MediaTailor

Fonctionnalité IAM	MediaTailor soutien
Identity-based politiques	Oui
Resource-based politiques	Oui
Actions de politique	Oui
Ressources de politique	Non
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
Rôles du service	Non
Service-linked rôles	Oui

Pour obtenir une vue d'ensemble de la façon dont MediaTailor et d'autres AWS services fonctionnent avec la plupart des fonctionnalités IAM, consultez la section [AWS Services qui fonctionnent avec IAM](#) dans le Guide de l'utilisateur IAM.

Identity-based politiques pour MediaTailor

Prend en charge les politiques basées sur l'identité : oui

Identity-based les politiques sont des documents de politique d'autorisations JSON que vous pouvez joindre 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 [Définition d'autorisations IAM personnalisées avec des politiques gérées par le client](#) 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. 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.

Identity-based exemples de politiques pour MediaTailor

Pour consulter des exemples de politiques MediaTailor basées sur l'identité, consultez. [Identity-based exemples de politiques pour AWS Elemental MediaTailor](#)

Resource-based politiques au sein de MediaTailor

Prend en charge les politiques basées sur les ressources : oui

Le MediaTailor service ne prend en charge qu'un seul type de politique basée sur les ressources. C'est ce qu'on appelle une politique de chaîne car elle est associée à une chaîne. Cette politique définit quels acteurs peuvent effectuer des actions sur la chaîne.

Resource-based les politiques sont des documents de stratégie JSON que vous joignez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. 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. Pour plus d'informations, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

Pour savoir comment associer une politique basée sur les ressources à une chaîne, consultez. [Création d'une chaîne à l'aide de la MediaTailor console](#)

Resource-based exemples de politiques dans MediaTailor

Pour consulter des exemples de politiques MediaTailor basées sur les ressources, consultez. [Resource-based exemples de politiques pour AWS Elemental MediaTailor](#)

Actions politiques pour MediaTailor

Prend en charge les actions de politique : oui

Les administrateurs peuvent utiliser des 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. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour consulter la liste des MediaTailor actions, voir [Actions définies par AWS Elemental MediaTailor](#) dans la Référence d'autorisation de service.

Les actions de stratégie en MediaTailor cours utilisent le préfixe suivant avant l'action :

```
mediatailor
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Pour consulter des exemples de politiques MediaTailor basées sur l'identité, consultez. [Identity-based exemples de politiques pour AWS Elemental MediaTailor](#)

Ressources politiques pour MediaTailor

Prend en charge les ressources de politique : non

AWS Elemental MediaTailor ne permet pas de spécifier des ARN de ressources dans une politique.

Clés de condition de politique pour MediaTailor

Prend en charge les clés de condition de politique spécifiques au service : oui

Les administrateurs peuvent utiliser des 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` indique à quel moment les instructions s'exécutent en fonction de critères définis. 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. Pour voir toutes les clés de condition AWS globales, consultez les clés contextuelles de conditions [AWS globales](#) dans le guide de l'utilisateur IAM.

Pour une liste des clés de MediaTailor condition, voir Clés de [condition pour AWS Elemental MediaTailor](#) dans la référence d'autorisation de service. Pour savoir avec quelles actions et quelles ressources vous pouvez utiliser une clé de condition, consultez la section [Actions définies par AWS Elemental MediaTailor](#).

AWS Elemental MediaTailor ne fournit pas de clés de condition spécifiques au service, mais prend en charge l'utilisation de certaines clés de condition globales. Pour voir toutes les clés de condition AWS globales, consultez la section Clés contextuelles de condition [AWS globale](#) dans le Guide de Gestion des identités et des accès AWS l'utilisateur.

ACL dans MediaTailor

Prend en charge les ACL : non

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.

ABAC avec MediaTailor

Prend en charge ABAC (identifications dans les politiques) : partiellement

Attribute-based le contrôle d'accès (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction d'attributs appelés balises. Vous pouvez associer des balises aux entités et AWS ressources IAM, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à celle de la ressource.

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 ABAC, consultez [Définition d'autorisations avec l'autorisation 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 MediaTailor, utilisez la valeur Partiel.

Utilisation d'informations d'identification temporaires avec MediaTailor

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès de courte durée aux AWS ressources et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. 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](#) et [Services AWS compatibles avec IAM](#) dans le Guide de l'utilisateur IAM.

Cross-service autorisations principales pour MediaTailor

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions d'accès direct (FAS) utilisent les autorisations du principal appelant et Service AWS, combinées Service AWS à la demande d'envoi de demandes aux services en aval. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez la section [Sessions de transmission d'accès](#).

Rôles de service pour MediaTailor

Prend en charge les rôles de service : Non

AWS Elemental MediaTailor ne prend pas en charge les rôles de service.

Service-linked rôles pour MediaTailor

Prend en charge les rôles liés à un service : oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut assumer le rôle d'effectuer une action en votre nom. Service-linked les rôles apparaissent dans votre compte Compte AWS et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus d'informations sur la création ou la gestion de rôles MediaTailor liés à un service, consultez. [Utilisation de rôles liés à des services pour MediaTailor](#)

Identity-based exemples de politiques pour AWS Elemental MediaTailor

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou modifier les ressources MediaTailor. 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.

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 IAM \(console\)](#) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par MediaTailor, y compris le format des ARN pour chacun des types de ressources, voir [Actions, ressources et clés de condition pour AWS Elemental MediaTailor](#) dans la référence d'autorisation de service.

Rubriques

- [Bonnes pratiques en matière de politiques](#)
- [Utilisation de la MediaTailor console](#)
- [Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations](#)

Bonnes pratiques en matière de politiques

Identity-based les politiques déterminent si quelqu'un peut créer des MediaTailor ressources sur votre compte, y accéder ou en supprimer. 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 :

- Commencez AWS par les politiques gérées et passez aux autorisations avec le moindre privilège — Pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par le AWS client qui sont

spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez [politiques gérées par AWS](#) ou [politiques gérées par AWS pour les activités professionnelles](#) dans le Guide de l'utilisateur IAM.

- Accordez 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 d'IAM pour appliquer des autorisations, consultez [politiques et autorisations dans IAM](#) dans le Guide de l'utilisateur IAM.
- Utilisez 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 par le biais d'un outil spécifique Service AWS, tel que CloudFormation. Pour plus d'informations, consultez [Conditions pour éléments de politique JSON IAM](#) dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM 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 plus d'informations, consultez [Validation de politiques avec IAM Access Analyzer](#) dans le Guide de l'utilisateur IAM.
- Exiger une authentification multifacteur (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root Compte AWS, activez l'authentification multifacteur pour renforcer la sécurité. Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez [Sécurisation de l'accès aux API avec 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 de la MediaTailor console

Pour accéder à la AWS Elemental MediaTailor console, vous devez disposer d'un ensemble minimum d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails MediaTailor des ressources de votre Compte AWS. Si vous créez une politique 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 politique.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour vous assurer que les utilisateurs et les rôles peuvent toujours utiliser la MediaTailor console, attachez également la politique MediaTailor *ConsoleAccess* ou la politique *ReadOnly* AWS gérée aux entités. Pour plus d'informations, consultez [Ajout d'autorisations à un utilisateur](#) dans le Guide de l'utilisateur IAM.

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 des autorisations pour effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

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

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

Resource-based exemples de politiques pour AWS Elemental MediaTailor

Pour savoir comment associer une politique basée sur les ressources à une chaîne, consultez.

[Création d'une chaîne à l'aide de la MediaTailor console](#)

Rubriques

- [Accès anonyme](#)
- [Cross-account accès](#)

Accès anonyme

Tenez compte de la Allow politique suivante. Lorsque cette politique est en vigueur, MediaTailor autorise un accès anonyme à l'`mediatailor:GetManifestation` sur la ressource de canal dans la politique. Cela se produit lorsque *region* se trouvent le Région AWS, *accountID* votre Compte AWS identifiant et *channelName* le nom de la ressource de chaîne.

Cross-account accès

Tenez compte de la Allow politique suivante. Lorsque cette politique est en vigueur, MediaTailor autorise l'`mediatailor:GetManifestation` sur la ressource du canal dans la politique, pour tous les comptes. Cela se produit lorsque *region* se trouvent le Région AWS, *accountID* votre Compte AWS identifiant et *channelName* le nom de la ressource de chaîne.

AWS politiques gérées pour AWS Elemental MediaTailor

Pour ajouter des autorisations aux utilisateurs, aux groupes et aux rôles, il est plus facile d'utiliser des politiques AWS gérées que de rédiger vous-même des politiques. Il faut du temps et de l'expertise pour [créer des politiques gérées par le client IAM](#) qui ne fournissent à votre équipe que les autorisations dont elle a besoin. Pour démarrer rapidement, vous pouvez utiliser nos politiques AWS gérées. Ces politiques couvrent des cas d'utilisation courants et sont disponibles dans votre Compte AWS. Pour plus d'informations sur les politiques AWS gérées, consultez les politiques [AWS gérées](#) dans le Guide de l'utilisateur IAM.

AWS les services maintiennent et mettent à jour les politiques AWS gérées. Vous ne pouvez pas modifier les autorisations dans les politiques AWS gérées. Les services ajoutent occasionnellement des autorisations à une politique gérée par AWS pour prendre en charge de nouvelles fonctionnalités. Ce type de mise à jour affecte toutes les identités (utilisateurs, groupes et rôles) auxquelles la politique est attachée. Les services sont très susceptibles de mettre à jour une politique gérée par AWS quand une nouvelle fonctionnalité est lancée ou quand de nouvelles opérations sont disponibles. Les services ne suppriment pas les autorisations d'une politique AWS gérée, de sorte que les mises à jour des politiques n'annuleront pas vos autorisations existantes.

En outre, AWS prend en charge les politiques gérées pour les fonctions qui couvrent plusieurs services. Par exemple, la politique `ReadOnlyAccess` AWS gérée fournit un accès en lecture seule à tous les AWS services et ressources. Lorsqu'un service lance une nouvelle fonctionnalité, AWS ajoute des autorisations en lecture seule pour les nouvelles opérations et ressources. Pour obtenir la liste des politiques de fonctions professionnelles et leurs descriptions, consultez la page [politiques gérées par AWS pour les fonctions de tâche](#) dans le Guide de l'utilisateur IAM.

AWS stratégie gérée : `AWSElementalMediaTailorFullAccess`

Vous pouvez associer la politique `AWSElementalMediaTailorFullAccess` à vos identités IAM. Il est utile pour les utilisateurs qui ont besoin de créer et de gérer des configurations de lecture et des ressources d'assemblage de canaux, telles que des programmes et des chaînes. Cette politique accorde des autorisations qui permettent un accès complet à AWS Elemental MediaTailor. Ces utilisateurs peuvent créer, mettre à jour et supprimer MediaTailor des ressources.

Pour voir les autorisations de cette stratégie, consultez [AWSElementalMediaTailorFullAccess](#) dans le AWS Guide de référence des stratégies gérées par.

AWS stratégie gérée : `AWSElementalMediaTailorReadOnly`

Vous pouvez associer la politique `AWSElementalMediaTailorReadOnly` à vos identités IAM. Il est utile pour les utilisateurs qui ont besoin de visualiser les configurations de lecture et les

ressources d'assemblage de canaux, telles que les programmes et les chaînes. Cette politique accorde des autorisations qui autorisent l'accès en lecture seule à AWS Elemental MediaTailor. Ces utilisateurs ne peuvent pas créer, mettre à jour ou supprimer des ressources MediaTailor.

Pour voir les autorisations de cette stratégie, consultez [AWSElementalMediaTailorReadOnly](#) dans le AWS Guide de référence des stratégies gérées par.

MediaTailor mises à jour de AWS stratégies gérées

Consultez les détails des mises à jour apportées aux politiques AWS gérées MediaTailor depuis que ce service a commencé à suivre ces modifications. Pour obtenir des alertes automatiques concernant les modifications apportées à cette page, abonnez-vous au flux RSS de la page MediaTailor [Historique des documents pour AWS Elemental MediaTailor](#).

Modifier	Description	Date
MediaTailor ajout de nouvelles politiques gérées	MediaTailor a ajouté les politiques gérées suivantes : • AWSElementalMediaTailorReadOnly • AWSElementalMediaTailorFullAccess	24 novembre 2021
MediaTailor a commencé à suivre les modifications	MediaTailor a commencé à suivre les modifications apportées AWS à ses politiques gérées.	24 novembre 2021

Utilisation de rôles liés à des services pour MediaTailor

AWS Elemental MediaTailor utilise des Gestion des identités et des accès AWS rôles liés à des [services \(IAM\)](#). Un rôle lié à un service est un type unique de rôle IAM directement lié à MediaTailor Service-linked les rôles sont prédéfinis MediaTailor et incluent toutes les autorisations dont le service a besoin pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service facilite la configuration MediaTailor car vous n'avez pas à ajouter manuellement les autorisations nécessaires. MediaTailor définit les autorisations de ses rôles liés

aux services et, sauf indication contraire, seule MediaTailor peut assumer ses rôles. Les autorisations définies comprennent la politique de confiance et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Vous pouvez supprimer un rôle lié à un service uniquement après la suppression préalable de ses ressources connexes. Cela protège vos MediaTailor ressources car vous ne pouvez pas supprimer par inadvertance l'autorisation d'accès aux ressources.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés aux services, consultez la section [AWS Services qui fonctionnent avec IAM](#) et recherchez les services dont la mention Oui est indiqué dans la Service-Linked colonne Rôle. Choisissez un Oui ayant un lien permettant de consulter les détails du rôle pour ce service.

Régions prises en charge pour les MediaTailor rôles liés aux services

MediaTailor prend en charge l'utilisation de rôles liés aux services dans toutes les régions où le service est disponible. Pour plus d'informations, consultez [AWS Régions et points de terminaison](#).

Rubriques

- [Service-linked autorisations de rôle pour MediaTailor](#)
- [Création d'un rôle lié à un service pour MediaTailor](#)
- [Modifier un rôle lié à un service pour MediaTailor](#)
- [Supprimer un rôle lié à un service pour MediaTailor](#)

Service-linked autorisations de rôle pour MediaTailor

MediaTailor utilise le rôle lié à un service nommé `AWSServiceRoleForMediaTailor` : MediaTailor utilise ce rôle lié à un service pour créer et gérer des groupes de CloudWatch journaux, des flux de journaux et des événements de journal. Ce rôle lié à un service est attaché à la politique gérée suivante : `AWSMediaTailorServiceRolePolicy`.

Le rôle `AWSServiceRoleForMediaTailor` lié à un service fait confiance aux services suivants pour assumer le rôle :

- `mediatailor.amazonaws.com`

La politique d'autorisation des rôles MediaTailor permet d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `logs:PutLogEvents` sur `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*:log-stream:*`
- Action : `logs:CreateLogStream`, `logs:CreateLogGroup`, `logs:DescribeLogGroups`, `logs:DescribeLogStreams` sur `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*`

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez la section Autorisations de [Service-Linked rôle](#) dans le guide de l'utilisateur IAM.

Création d'un rôle lié à un service pour MediaTailor

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous activez la journalisation de session dans l'API Console de gestion AWS, le AWS Command Line Interface (AWS CLI) ou l' AWS API MediaTailor crée le rôle lié au service pour vous.

Important

Ce rôle lié à un service peut apparaître dans votre compte si vous avez effectué une action dans un autre service qui utilise les fonctions prises en charge par ce rôle. De plus, si vous utilisiez le MediaTailor service avant le 15 septembre 2021, date à laquelle il a commencé à prendre en charge les rôles liés aux services, vous avez MediaTailor créé le `AWSServiceRoleForMediaTailor` rôle dans votre compte. Pour plus d'informations, consultez [A New Role Appeared in My IAM Account](#) (Un nouveau rôle est apparu dans mon compte IAM).

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous activez la journalisation des sessions, le rôle lié au service est MediaTailor créé à nouveau pour vous.

Vous pouvez également utiliser la console IAM pour créer un rôle lié à un service avec le MediaTailor cas d'utilisation. Dans l'API AWS CLI ou l' AWS API, créez un rôle lié à un service avec le nom du `mediatailor.amazonaws.com` service. Pour plus d'informations, consultez la section [Création d'un Service-Linked rôle](#) dans le guide de l'utilisateur IAM. Si vous supprimez ce rôle lié à un service, vous pouvez utiliser ce même processus pour créer le rôle à nouveau.

Modifier un rôle lié à un service pour MediaTailor

MediaTailor ne vous permet pas de modifier le rôle `AWSServiceRoleForMediaTailor` lié au service. Après avoir créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez la section [Modifier un Service-Linked rôle](#) dans le guide de l'utilisateur IAM.

Supprimer un rôle lié à un service pour MediaTailor

Si vous n'avez plus besoin d'utiliser une fonctionnalité ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité inutilisée qui n'est pas surveillée ou gérée activement. Cependant, vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.

Note

Si le MediaTailor service utilise le rôle lorsque vous essayez de nettoyer les ressources, la suppression risque d'échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour nettoyer MediaTailor les ressources utilisées par `AWSServiceRoleForMediaTailor`

- Avant de pouvoir supprimer le rôle lié à un service créé par MediaTailor pour la configuration du journal, vous devez d'abord désactiver toutes les configurations de journal de votre compte. Pour désactiver une configuration de journal, définissez la valeur du pourcentage activé sur 0. Cela désactive toutes les sessions qui enregistrent la configuration de lecture correspondante. Pour de plus amples informations, veuillez consulter [Désactivation de la configuration d'un journal](#).

Pour supprimer manuellement le rôle lié au service à l'aide d'IAM

Utilisez la console IAM, le AWS Command Line Interface (AWS CLI) ou l' AWS API pour supprimer le rôle lié à un `AWSServiceRoleForMediaTailor` service. Pour plus d'informations, consultez [la section Supprimer un Service-Linked rôle](#) dans le guide de l'utilisateur IAM.

Résolution des problèmes AWS Elemental MediaTailor identité et accès

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pouvez rencontrer lors de l'utilisation MediaTailor d'un IAM.

Rubriques

- [Je ne suis pas autorisé à effectuer une action dans MediaTailor](#)
- [Je ne suis pas autorisé à effectuer iam : PassRole](#)
- [Je souhaite autoriser des personnes extérieures à mon Compte AWS pour accéder à mes MediaTailor ressources](#)
- [MediaTailor Impossible d'accéder à mon compartiment Amazon S3](#)

Je ne suis pas autorisé à effectuer une action dans MediaTailor

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 `mediatailor:GetWidget` fictives.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
mediatailor: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 `mediatailor: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 MediaTailor.

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, vous devez 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 MediaTailor. Toutefois, l'action nécessite que le service ait des autorisations accordées par une fonction de service. Mary n'est pas autorisée à transmettre 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 autoriser des personnes extérieures à mon Compte AWS pour accéder à mes MediaTailor 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 plus d'informations, consultez les éléments suivants :

- Pour savoir si ces fonctionnalités sont prises MediaTailor en charge, consultez [Comment ? AWS Elemental MediaTailor fonctionne avec IAM](#).
- Pour savoir comment donner accès à vos ressources par l'intermédiaire de celles Comptes AWS que vous possédez, consultez la section [Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS qui vous appartient](#) dans le Guide de l'utilisateur IAM.
- Pour savoir comment donner 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 en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

MediaTailor Impossible d'accéder à mon compartiment Amazon S3

Si des MediaTailor erreurs s'affichent lorsque vous tentez d'accéder au contenu de votre compartiment Amazon S3, vérifiez les causes courantes suivantes :

Problèmes liés aux autorisations Amazon S3

Symptômes d'erreur :

- 502 BadGatewayException avec des messages relatifs à Amazon S3
- 403 Forbidden erreurs dans les MediaTailor journaux
- Le contenu ne se charge pas depuis l'origine Amazon S3

Résolution :

1. Vérifiez que votre rôle de MediaTailor service dispose des autorisations Amazon S3 requises :
2. Vérifiez que votre politique relative aux compartiments Amazon S3 autorise MediaTailor l'accès :

Problèmes d'accès HTTPS

Symptômes d'erreur :

- SSL/TLS échecs de poignée de main
- 502 Bad Gateway lors de l'accès à des URL HTTPS Amazon S3

Résolution :

1. Assurez-vous que votre politique relative aux compartiments Amazon S3 nécessite le protocole HTTPS :

2. Vérifiez que MediaTailor la configuration utilise des URL HTTPS pour le contenu Amazon S3.

Cross-account Accès à Amazon S3

Symptômes d'erreur :

- Access Denied erreurs lorsque le compartiment Amazon S3 se trouve dans un autre compte
- MediaTailor ne peut pas assumer de rôles multicomptes

Résolution :

1. Mettez à jour la politique de compartiment Amazon S3 dans le compte de contenu en fonction du rôle MediaTailor de service de Trust
2. Ajouter des autorisations multicomptes au rôle MediaTailor de service
3. Assurez-vous que les relations de confiance appropriées sont configurées pour les deux comptes

Validation de conformité pour AWS Elemental MediaTailor

Notre nouvelle expérience AWS d'inscription n'est pas conçue pour les charges de travail réglementées. Si vous utilisez notre nouvelle interface d' AWS inscription, mais que vous souhaitez l'utiliser AWS pour des charges de travail réglementées, vous pouvez vous [inscrire à des fonctionnalités AWS \(avancées\)](#) ou [activer des fonctionnalités avancées](#) pour votre AWS environnement.

Pour savoir si un s' Service AWS inscrit dans le cadre de programmes de conformité spécifiques, consultez [Services AWS la section Étendue par programme de conformité](#) et choisissez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de [AWS conformité Programmes](#) .

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 dépend de la sensibilité de vos données, des objectifs de conformité de votre entreprise et des lois et réglementations applicables. Pour plus d'informations sur votre responsabilité en matière de conformité lors de l'utilisation Services AWS, consultez [AWS la documentation de sécurité](#).

Résilience dans AWS Elemental MediaTailor

L'infrastructure AWS mondiale est construite autour Régions AWS de zones de disponibilité. Régions AWS fournissent plusieurs zones de disponibilité physiquement séparées et isolées, connectées à un réseau à faible latence, haut débit et hautement redondant. 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 Régions AWS les zones de disponibilité, consultez la section Infrastructure [AWS globale](#).

Outre l'infrastructure AWS mondiale, il MediaTailor propose plusieurs fonctionnalités pour répondre à vos besoins en matière de résilience et de sauvegarde des données.

Sécurité de l'infrastructure dans MediaTailor

En tant que service géré, AWS Elemental MediaTailor il est protégé par la sécurité du réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et la manière de AWS protéger l'infrastructure, consultez [AWS Cloud Security](#). Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de [l'infrastructure](#) dans Security Pillar AWS Well-Architected Framework.

Vous utilisez des appels d'API AWS publiés pour accéder MediaTailor via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Suites de chiffrement avec une confidentialité directe parfaite (PFS) telles que DHE (Ephemeral) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

Vous pouvez appeler ces opérations d'API depuis n'importe quel emplacement réseau, mais MediaTailor elles prennent en charge les politiques d'accès basées sur les ressources, qui peuvent inclure des restrictions basées sur l'adresse IP source. Vous pouvez également utiliser des MediaTailor politiques pour contrôler l'accès à partir de points de terminaison Amazon Virtual Private Cloud (Amazon VPC) spécifiques ou de VPC spécifiques. En fait, cela isole l'accès réseau à une MediaTailor ressource donnée du seul VPC spécifique du AWS réseau.

Cross-service prévention adjointe confuse

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion du problème des adjoints. Cross-service l'usurpation d'identité peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé à accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte.

Nous vous recommandons d'utiliser les clés contextuelles de condition SourceAccount globale https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_condition-keys.html#condition-keys-sourcearn `aws : SourceArn` et `aws :` dans les politiques relatives aux ressources afin de limiter les autorisations qui AWS Elemental MediaTailor accordent un autre service à la ressource. Si vous utilisez les deux clés de contexte de condition globale, la valeur `aws : SourceAccount` et le compte de la valeur `aws : SourceArn` doit utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de stratégie.

La valeur de `aws : SourceArn` doit correspondre à la configuration de lecture pour laquelle MediaTailor les CloudWatch journaux sont publiés dans votre région et votre compte. Toutefois, cela ne s'applique que si vous utilisez le [MediaTailorLogger](#) rôle qui permet de MediaTailor publier CloudWatch les journaux Amazon sur votre compte. Cela ne s'applique pas si vous utilisez un rôle [lié à un service](#) pour autoriser la MediaTailor CloudWatch publication des journaux.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws : SourceArn` avec l'ARN complet de la ressource. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws : SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:service: :123456789012:*`.

L'exemple suivant montre comment vous pouvez utiliser les touches contextuelles `aws : SourceArn` et les clés de contexte de la condition `aws : SourceAccount` globale MediaTailor pour éviter le problème de confusion entre les sous-ministres.

Enregistrement et surveillance dans MediaTailor

Cette section fournit une présentation des options de consignation et surveillance dans AWS Elemental MediaTailor à des fins de sécurité. Pour plus d'informations sur la journalisation et la surveillance, MediaTailor voir [Surveillance et balisage AWS Elemental MediaTailor resources](#).

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité AWS Elemental MediaTailor et des performances de vos AWS solutions. Vous devez collecter les données de surveillance de toutes les parties de votre AWS solution afin de pouvoir déboguer plus facilement une panne multipoint si elle se produit. AWS fournit plusieurs outils pour surveiller vos MediaTailor ressources et répondre aux incidents potentiels :

CloudWatch Alarmes Amazon

À l'aide d' CloudWatch alarmes, vous observez une seule métrique sur une période que vous spécifiez. Si la métrique dépasse un seuil donné, une notification est envoyée à une rubrique Amazon SNS ou à une politique AWS Auto Scaling. CloudWatch les alarmes n'invoquent pas d'actions parce qu'elles se trouvent dans un état particulier. L'état doit avoir changé et avoir été conservé pendant un nombre de périodes spécifié. Pour de plus amples informations, veuillez consulter [the section called "Surveillance à l'aide de CloudWatch métriques"](#).

AWS CloudTrail journaux

CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans AWS Elemental MediaTailor. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite MediaTailor, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des informations supplémentaires. Pour de plus amples informations, veuillez consulter [Enregistrement AWS Elemental MediaTailor Appels d'API](#).

AWS Trusted Advisor

Trusted Advisor s'appuie sur les meilleures pratiques apprises au service de centaines de milliers de AWS clients. Trusted Advisor inspecte votre AWS environnement, puis formule des recommandations lorsque des opportunités se présentent pour économiser de l'argent, améliorer la disponibilité et les performances du système ou contribuer à combler les failles de sécurité. Tous les clients AWS ont accès à cinq contrôles Trusted Advisor. Les clients avec un plan de support Business ou Enterprise peuvent afficher tous les contrôles Trusted Advisor .

Pour de plus amples informations, veuillez consulter [AWS Trusted Advisor](#).

Surveillance et balisage AWS Elemental MediaTailor

resources

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité AWS Elemental MediaTailor et des performances de vos autres AWS solutions. AWS fournit les outils de surveillance suivants pour surveiller MediaTailor, 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 de vos instances Amazon EC2 et lancer automatiquement de nouvelles instances si nécessaire. Pour plus d'informations, consultez le guide de CloudWatch l'utilisateur [Amazon](#).
- Amazon CloudWatch Logs vous permet de surveiller, de stocker et d'accéder à vos fichiers journaux à partir de toutes les interactions avec votre serveur de décision publicitaire (ADS). AWS Elemental MediaTailor émet des journaux pour les demandes d'annonces, les redirections, les réponses et les demandes de rapports et les réponses. Des erreurs provenant des serveurs ADS et d'origine sont également transmises aux groupes de journaux d'Amazon CloudWatch. MediaTailor fournit également des informations sur les publicités ignorées et les raisons pour lesquelles elles ont été ignorées. Pour de plus amples informations, veuillez consulter [Résolution des problèmes liés à l'omission d'annonces](#). Vous pouvez également archiver vos données de journaux dans une solution de stockage hautement durable. Pour des informations générales, consultez le guide de l'utilisateur [Amazon CloudWatch Logs](#). Pour plus d'informations sur les journaux ADS et sur la manière d'y accéder à des fins d'analyse via Amazon CloudWatch Logs Insights, consultez [AWS Elemental MediaTailor Analyse des journaux ADS dans Amazon CloudWatch Logs Insights](#).

Rubriques

- [Visualisation AWS Elemental MediaTailor journaux](#)
- [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#)
- [Enregistrement AWS Elemental MediaTailor Appels d'API](#)
- [Réception AWS Elemental MediaTailor alertes d'assemblage de canaux](#)

- [Identification AWS Elemental MediaTailor resources](#)
- [Contrôle AWS services multimédia avec moniteur de flux de travail](#)

Visualisation AWS Elemental MediaTailor journaux

MediaTailor émet des journaux qui décrivent une variété de jalons et d'activités concernant les canaux et les configurations de lecture. Vous pouvez utiliser ces journaux pour améliorer la visibilité de votre flux de travail et résoudre les problèmes liés au service. Les rubriques suivantes décrivent les journaux et les options de journalisation.

Rubriques

- [AWS Elemental MediaTailor Description des journaux ADS et types d'événements](#)
- [AWS Elemental MediaTailor description des journaux manifestes et types d'événements](#)
- [AWS Elemental MediaTailor description des journaux de transcodage et types d'événements](#)
- [Utiliser des journaux vendus pour envoyer AWS Elemental MediaTailor journaux](#)
- [Ecriture AWS Elemental MediaTailor se connecte directement à Amazon CloudWatch Logs](#)
- [Contrôler le volume de AWS Elemental MediaTailor journaux](#)
- [Filtrage AWS Elemental MediaTailor journaux et événements](#)
- [Génération AWS Elemental MediaTailor journaux de débogage](#)

AWS Elemental MediaTailor Description des journaux ADS et types d'événements

Les sections suivantes décrivent les journaux qui MediaTailor sont émis pour décrire les événements liés au serveur de décision publicitaire (ADS). Ce sont `AdDecisionServerInteractions` des journaux.

Rubriques

- [AdDecisionServerInteractions events](#)
- [Description du journal ADS](#)
- [Schéma JSON du journal ADS](#)

AdDecisionServerInteractions events

Les événements suivants sont émis lors MediaTailor des interactions avec le serveur de décision publicitaire (ADS).

Journal	Description
AD_MARKER_FOUND	MediaTailor J'ai trouvé un marqueur publicitaire dans le manifeste.
BEACON_FIRED	Une balise de suivi a été déclenchée pour signaler des événements publicitaires. En mode de création de rapports côté serveur (par défaut), MediaTailor déclenche la balise. En mode rapport côté client, l'appareil de lecture déclenche la balise.
EMPTY_VAST_RESPONSE	L'ADS a renvoyé une réponse VAST vide contenant aucune annonce.
EMPTY_VMAP_RESPONSE	L'ADS a renvoyé une réponse VMAP vide.
ERROR_ADS_INVALID_RESPONSE	L'ADS a renvoyé un code d'état différent de 200.
ERROR_ADS_IO	MediaTailor J'ai rencontré une erreur lors de la tentative de communication avec l'ADS.
ERROR_ADS_RESPONSE_PARSE	MediaTailor a rencontré une erreur lors de l'analyse de la réponse ADS.
ERROR_ADS_RESPONSE_UNKNOWN_ROOT_ELEMENT	La réponse ADS contient un élément racine non valide.
ERROR_ADS_TIMEOUT	La MediaTailor demande adressée à l'ADS a expiré.
ERROR_DISALLOWED_HOST	L'hébergeur ADS n'est pas autorisé.

Journal	Description
ERROR_FIRING_BEACON_FAILED	MediaTailor n'a pas réussi à déclencher la balise de repérage.
ERROR_PERSONALIZATION_DISABLED	L'insertion d'annonces est désactivée pour cette session.
ERROR_UNKNOWN	MediaTailor a rencontré une erreur inconnue lors de la demande ADS.
ERROR_UNKNOWN_HOST	L'hôte ADS est inconnu.
ERROR_VAST_INVALID_MEDIA_FILE	Le VAST Ad contient un MediaFile élément non valide ou manquant.
ERROR_VAST_INVALID_VAST_AD_TAG_URI	La réponse VAST contient une valeur non valide VASTAdTagURI .
ERROR_VAST_MISSING_CREATIVES	Le VAST Ad ne contient aucun élément ou plusieurs Creatives éléments. Exactement un Creatives élément est requis.
ERROR_VAST_MISSING_IMPRESSION	Le VAST Ad ne contient aucun Impression élément. Au moins un Impression élément est requis.
ERROR_VAST_MISSING_MEDIAFILES	Le VAST Ad ne contient aucun élément ou plusieurs MediaFiles éléments. Exactement un MediaFiles élément est requis.
ERROR_VAST_MISSING_OVERLAYS	MediaTailor n'a reçu aucune création non linéaire de la part du serveur publicitaire.
ERROR_VAST_MULTIPLE_LINEAR	Le VAST Ad contient de multiples Linear éléments.
ERROR_VAST_MULTIPLE_TRACKING_EVENTS	Le VAST Ad contient de multiples TrackingEvents éléments.

Journal	Description
ERROR_VAST_REDIRECT_EMPTY_RESPONSE	La demande de redirection VAST a renvoyé une réponse vide.
ERROR_VAST_REDIRECT_FAILED	La demande de redirection VAST a rencontré une erreur.
ERROR_VAST_REDIRECT_MULTIPLE_VAST	La demande de redirection VAST a renvoyé plusieurs annonces.
FILLED_AVAIL	MediaTailor a rempli le réservoir avec succès.
FILLED_OVERLAY_AVAIL	MediaTailor a correctement rempli la zone de superposition.
INTERSTITIAL_VOD_FAILURE	La demande ou la réponse ADS a rencontré un problème lors du remplissage des espaces interstitiels pour la playlist VOD. Aucune annonce ne sera insérée.
INTERSTITIAL_VOD_SUCCESS	MediaTailor zones interstitielles remplies avec succès pour la playlist VOD.
MAKING_ADS_REQUEST	MediaTailor demande des publicités à l'ADS.
MODIFIED_TARGET_URL	MediaTailor modifié l'URL cible sortante.
NON_AD_MARKER_FOUND	MediaTailor J'ai trouvé un marqueur publicitaire non exploitable dans le manifeste.
PRE_ADS_REQUEST_FUNCTION_COMPLETED	Une fonction individuelle du hook de demande préalable aux publicités est terminée. Il s'agit d'un type d'événement optionnel.
PRE_ADS_REQUEST_FUNCTION_ERROR	Une fonction individuelle du hook de demande préalable aux publicités a échoué.
PRE_ADS_REQUEST_HOOK_ERROR	L'exécution du hook de requête préalable à la publication a échoué.

Journal	Description
PRE_ADS_REQUEST_HOOK_SUMMARY	Résumé de l'exécution du hook de demande avant la publication, y compris le statut de réussite ou d'erreur. Il s'agit d'un type d'événement optionnel.
RAW_ADS_RESPONSE	MediaTailor a reçu une réponse ADS brute.
REDIRECTED_VAST_RESPONSE	MediaTailor a reçu une réponse VAST après avoir suivi la redirection VAST.
VAST_REDIRECT	La réponse à l'annonce VAST contient une redirection.
VAST_RESPONSE	MediaTailor a reçu une réponse VASTE.
VOD_TIME_BASED_AVAIL_PLAN_SUCCESS	MediaTailor a créé avec succès un plan de disponibilité basé sur le temps pour le modèle de VOD.
VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET	MediaTailor crée un plan de disponibilité basé sur le temps pour le modèle de VOD. MediaTailor a reçu une réponse VAST pour le décalage horaire.
VOD_TIME_BASED_AVAIL_PLAN_WARNING_NO_ADVERTISEMENTS	La demande ou la réponse ADS a rencontré un problème lors de la création d'un plan de disponibilité basé sur le temps pour le modèle de VOD. Aucune annonce ne sera insérée.
WARNING_NO_ADVERTISEMENTS	MediaTailor J'ai rencontré un problème lors du remplissage du réservoir. Aucune annonce n'est insérée.
WARNING_URL_VARIABLE_SUBSTITUTION_FAILED	MediaTailor Impossible de remplacer des variables dynamiques dans l'URL ADS. Vérifiez la configuration de l'URL.

Journal	Description
WARNING_VPAID_AD_DROPPED	Une annonce VPAID a été supprimée en raison d'une liste manquante, ou la session utilise des rapports côté serveur.

Description du journal ADS

Cette section décrit la structure et le contenu de la description du journal ADS. Pour explorer vous-même dans un éditeur JSON, utilisez la liste à l'adresse [the section called “Schéma JSON du journal ADS”](#).

Chaque événement du journal ADS contient les champs standard générés par CloudWatch Logs. Pour plus d'informations, voir [Analyser les données des CloudWatch journaux avec Logs Insights](#).

Propriétés des journaux ADS

Cette section décrit les propriétés des journaux ADS.

Propriété	Type	Obligatoire	Description
adsRequestUrl	chaîne	false	URL complète de la demande ADS effectuée par MediaTailor.
avail	objet de type avail	false	Informations sur une offre MediaTailor remplie d'annonces. Actuellement, pour le type d'FILLED_AVAIL événement, il s'agit du plan créé MediaTailor lors de sa première rencontre avec succès. La façon dont la diffusion est finalement remplie

Propriété	Type	Obligatoire	Description
			peut varier en fonction de la façon dont le contenu se déroule.
<code>awsAccountId</code>	chaîne	true	L'ID de compte AWS pour la MediaTailor configuration qui a été utilisée pour la session.
<code>customerId</code>	chaîne	true	Version hachée de l'ID de compte AWS, que vous pouvez utiliser pour établir une corrélation entre plusieurs entrées de journaux.
<code>eventDescription</code>	chaîne	true	Brève description de l'événement qui a déclenché ce message de journal, fournie par le MediaTailor service. Par défaut, celle-ci est vide. Exemple: Got VAST response.
<code>eventTimestamp</code>	chaîne	true	Date et heure de l'événement.
<code>eventType</code>	chaîne	true	Code de l'événement qui a déclenché ce message de journal. Exemple: VAST_RESPONSE .

Propriété	Type	Obligatoire	Description
<code>errorType</code>	chaîne	false	Classification structurée des erreurs. Présent sur PRE_ADS_R EQUEST_HO OK_ERROR et sur PRE_ADS_R EQUEST_FU NCTION_ERROR les événements. Valeurs :INTERNAL_ERROR , SYNTAX_ERROR RESOURCE_LIMIT_ERROR ,RESTRICTION_ERROR ,TIMEOUT_ERROR ,VALIDATION_ERROR .
<code>cause</code>	chaîne	false	Chaîne de détail de l'erreur décrivant la panne spécifique. Présent sur PRE_ADS_R EQUEST_HO OK_ERROR et sur PRE_ADS_R EQUEST_FU NCTION_ERROR les événements.

Propriété	Type	Obligatoire	Description
<code>eventId</code>	chaîne	false	Identifiant unique par appel de hook. Utilisez ce champ pour corréler les événements au niveau du hook et au niveau de la fonction pour la même exécution. Présent uniquement lors des événements Functions.
<code>executionTimeMs</code>	entier	false	Temps d'exécution en millisecondes. Présent uniquement lors des événements Functions.
<code>functionId</code>	chaîne	false	Identifiant de la fonction. Sur les événements hook, il s'agit de la fonction configurée sur le hook. En ce qui concerne les événements de fonction, il s'agit de la fonction individuelle au sein de l'exécuteur. Présent uniquement lors des événements Functions.

Propriété	Type	Obligatoire	Description
<code>functionType</code>	chaîne	false	Type de fonction :CUSTOM_OUTPUT , HTTP_REQUEST_SEQUENTIAL_EXECUTOR , ouCONCURRENT_EXECUTOR . Présent uniquement lors des événements au niveau de la fonction.
<code>http</code>	objet	false	Détails de la requête et de la réponse HTTP. Présent sur les événements au niveau des HTTP_REQUEST fonctions uniquement. Contient request (URL, méthode, en-têtes, corps),, response (StatusCode,responseTimeMs , en-têtes, corps) et truncated (vrai lorsque le corps de la demande ou de la réponse a été tronqué pour respecter les limites de taille des événements du journal).

Propriété	Type	Obligatoire	Description
input	objet	false	Before-values de variables ciblées par les expressions de sortie. Présent uniquement lors des événements au niveau de la fonction.
output	objet	false	Valeurs produites par la fonction. Présent uniquement PRE_ADS_REQUEST_COMPLETED lors d'événements.
status	chaîne	false	Résultat de l'exécution du crochet : SUCCESS ou ERROR. Présent uniquement PRE_ADS_REQUEST_HOOK_OK_SUMMARY lors d'événements.
originId	chaîne	true	Le nom de configuration indiqué dans la MediaTailor configuration. Cette source est différente de la source de contenu vidéo, qui fait également partie de la configuration.

Propriété	Type	Obligatoire	Description
<code>prefetchScheduleName</code>	chaîne	false	Nom du calendrier de prélecture associé à cet événement publicitaire.
<code>requestHeaders</code>	tableau de type requestheaders	false	Les en-têtes MediaTailor inclus dans la demande ADS. Généralement, les journaux incluent ceux-ci lorsqu'une demande à ADS échoue, pour faciliter le dépannage.
<code>requestId</code>	chaîne	true	L'ID de MediaTailor demande, que vous pouvez utiliser pour corréler plusieurs entrées de journal pour la même demande.

Propriété	Type	Obligatoire	Description
<code>sessionId</code>	chaîne	true	L'identifiant numérique unique MediaTailor attribué à la session du joueur. Toutes les demandes effectuées par un joueur pour une session ont le même ID de session. Exemple: e039fd39-09f0-46b2-aca9-9871cc116cde .
<code>sessionType</code>	string (valeurs légales : [DASH, HLS])	true	Type de flux du lecteur.
<code>vastAd</code>	objet de type vastAd	false	Informations sur une seule publicité analysée à partir de la réponse VAST.
<code>vastResponse</code>	objet de type vastResponse	false	Informations sur la réponse VAST MediaTailor reçue de l'ADS.

Propriété	Type	Obligatoire	Description
<code>vodCreativeOffsets</code>	objet de type vodCreativeOffsets	false	Une carte qui indique les décalages temporels dans le manifeste où les informations MediaTailor seront insérées, en fonction de la réponse VMAP.
<code>vodVastResponseTimeOffset</code>	number	false	Décalage horaire spécifique au VMAP pour l'insertion de publicités VOD.

adContent

Cette section décrit les propriétés de l'objet adContent des journaux ADS.

Propriétés de l'objet adContent des journaux ADS

Propriété	Type	Obligatoire	Description
<code>adPlaylistUri</code>	objet de type annoncePlaylistUri	false	Mappage du manifeste d'origine pour une variante au manifeste publicitaire pour la variante. Pour DASH, ceci contient une seule entrée, car toutes les variantes sont représentées dans un seul manifeste DASH.

annonce PlaylistUri

Cette section décrit les propriétés de l'annonce ADS logsPlaylistUri.

Logs et PlaylistUri propriétés ADS

Propriété	Type	Obligatoire	Description
<any string>	chaîne	false	URL du manifeste publicitaire pour la variante spécifique.

avail

Cette section décrit les propriétés de l'objet avail des journaux ADS.

Propriétés de l'objet avail des journaux ADS

Propriété	Type	Obligatoire	Description
availId	chaîne	true	Identifiant unique pour cette diffusion. Pour HLS, il s'agit du numéro de séquence média où la diffusion commence. Pour DASH, il s'agit de l'ID de période.
creativeAds	tableau de type creativeAd	true	Les annonces qui ont MediaTailor été insérées dans la liste.
fillRate	number	true	Vitesse à laquelle les publicités remplissent la durée de diffusion , de 0,0 (pour 0 %) à 1,0 (pour 100 %).

Propriété	Type	Obligatoire	Description
<code>filledDuration</code>	number	true	Somme des durées de toutes les publicités insérées dans la diffusion.
<code>numAds</code>	number	true	Le nombre d'annonces MediaTailor insérées dans la liste.
<code>originAvailabilityDuration</code>	number	true	Durée de la diffusion telle que spécifiée dans le flux de contenu à partir de l'origine (CUE_OUT ou SCTE).
<code>skippedAds</code>	tableau de type skippedAd	false	Les publicités que MediaTailor n'ont pas été insérées, pour des raisons telles que TRANSCODE_IN_PROGRESS et TRANSCODE_ERROR . Pour obtenir la liste des raisons pour lesquelles les annonces ont été ignorées, consultez Résolution des problèmes liés à l'omission d'annonces.

Propriété	Type	Obligatoire	Description
<code>slateAd</code>	objet de type slateAd	true	Informations sur l'annonce ardoise, qui permet MediaTailor de remplir tous les segments non remplis au fur et à mesure.

creativeAd

Cette section décrit les propriétés de l'objet creativeAd des journaux ADS.

Propriétés de l'objet creativeAd des journaux ADS

Propriété	Type	Obligatoire	Description
<code>adContent</code>	objet de type adContent	true	Informations sur le contenu de la publicité insérée.
<code>creativeUniqueId</code>	chaîne	true	Identifiant unique de la publicité, utilisé comme clé pour le transcodage. Il s'agit du champ ID de la création dans la réponse VAST, si disponible. Sinon, il s'agit de l'URL mezzanine de la publicité.
<code>trackingEvents</code>	objet de type trackingEvents	true	URL de balise de suivi pour les différents événements de suivi de la publicité. Les clés sont les noms

Propriété	Type	Obligatoire	Description
			d'événements et les valeurs sont une liste d'URL de balise.
transcodeAdDuration	number	true	Durée de la publicité, calculée à partir de l'asset transcodé.
uri	chaîne	true	URL de la version mezzanine de la publicité, qui est l'entrée du transcodeur.
vastDuration	number	true	Durée de la publicité, telle qu'analysée à partir de la réponse VAST.

requestheaders

Cette section décrit les propriétés des objets requestheaders des journaux ADS.

Propriétés des objets requestheaders des journaux ADS

Propriété	Type	Obligatoire	Description
name	chaîne	true	Nom de l'en-tête.
value	chaîne	true	Valeur de l'en-tête.

skippedAd

Cette section décrit les propriétés de l'objet skippedAd des journaux ADS.

Propriétés de l'objet skippedAd des journaux ADS

Propriété	Type	Obligatoire	Description
adMezzanineUrl	chaîne	true	URL mezzanine de la publicité ignorée.
creativeUniqueId	chaîne	true	Identifiant unique de la publicité, utilisé comme clé pour le transcodage. Il s'agit du champ ID de la création dans la réponse VAST, si disponible. Sinon, il s'agit de l'URL mezzanine de la publicité.
skippedReason	chaîne	true	Code qui indique pourquoi la publicité n'a pas été insérée. Exemple: TRANSCODE_IN_PROGRESS . Pour obtenir la liste des raisons pour lesquelles les annonces ont été ignorées, consultez Résolution des problèmes liés à l'omission d'annonces.
transcodeAdDuration	number	false	Durée de la publicité , calculée à partir de l'asset transcodé.

Propriété	Type	Obligatoire	Description
<code>vastDuration</code>	number	true	Durée de la publicité , telle qu'analysée à partir de la réponse VAST.

plateAd

Cette section décrit les propriétés de l'objet plateAd des journaux ADS.

Propriétés de l'objet plateAd des journaux ADS

Propriété	Type	Obligatoire	Description
<code>adContent</code>	objet de type adContent	true	Informations sur le contenu de la publicité insérée.
<code>creativeUniqueId</code>	chaîne	true	Identifiant unique de la publicité, utilisé comme clé pour le transcodage. Il s'agit du champ ID de la création dans la réponse VAST, si disponible. Sinon, il s'agit de l'URL mezzanine de la publicité.
<code>transcodeAdDuration</code>	number	true	Durée de la publicité , calculée à partir de l'asset transcodé.
<code>uri</code>	chaîne	true	URL de la version mezzanine de la publicité, qui est

Propriété	Type	Obligatoire	Description
			l'entrée du transcodeur.

trackingEvents

Cette section décrit les propriétés des objets trackingEvents des journaux ADS.

Propriétés des objets trackingEvents des journaux ADS

Propriété	Type	Obligatoire	Description
<any string>	tableau de type chaîne	false	Liste des URL de balise pour l'événement de suivi spécifié (impression, complet, etc.)

vastAd

Cette section décrit les propriétés de l'objet vastAd des journaux ADS.

Propriétés de l'objet vastAd des journaux ADS

Propriété	Type	Obligatoire	Description
adSystem	chaîne	true	Valeur de la balise AdSystem dans la réponse VAST.
adTitle	chaîne	true	Fichiers multimédias qui sont disponibles pour la publicité dans la réponse VAST.
creativeAdId	chaîne	true	Valeur de l'attribut adId de la balise

Propriété	Type	Obligatoire	Description
			Creative dans la réponse VAST.
creativeId	chaîne	true	Valeur de l'attribut id de la balise Creative dans la réponse VAST.
duration	number	true	Durée approximative de la publicité, en fonction de la balise duration dans l'élément linear de la réponse VAST.
trackingEvents	objet de type trackingEvents	true	URL de balise de suivi pour les différents événements de suivi de la publicité. Les clés sont les noms d'événements et les valeurs sont une liste d'URL de balise.
vastAdId	chaîne	true	Valeur de l'attribut id de la balise Ad dans la réponse VAST.
vastAdTagUri	chaîne	false	L'URI de VMAP-specific redirection d'une annonce.

Propriété	Type	Obligatoire	Description
<code>vastMediaFiles</code>	tableau de type vaste MediaFile	true	Liste des fichiers multimédias disponibles pour la publicité dans la réponse VAST.

vaste MediaFile

Cette section décrit en détail les propriétés des journaux ADSMediaFile.

ADS enregistre de vastes MediaFile propriétés

Propriété	Type	Obligatoire	Description
<code>apiFramework</code>	chaîne	true	Cadre d'API nécessaire pour gérer le fichier multimédia. Exemple: VPAID.
<code>bitrate</code>	number	true	Vitesse de transmission du fichier multimédia.
<code>delivery</code>	chaîne	true	Protocole utilisé pour le fichier multimédia, défini sur progressif ou streaming.
<code>height</code>	number	true	Hauteur en pixels du fichier multimédia.
<code>id</code>	chaîne	true	Valeur de l'attribut ID de la balise MediaFile .
<code>type</code>	chaîne	true	Type MIME du fichier multimédia, extrait de

Propriété	Type	Obligatoire	Description
			l'attribut de type de la balise MediaFile .
uri	chaîne	true	URL de la version mezzanine de la publicité, qui est l'entrée du transcodeur.
width	number	true	Largeur en pixels du fichier multimédia.

vastResponse

Cette section décrit les propriétés de l'objet vastResponse des journaux ADS.

Propriétés de l'objet vastResponse des journaux ADS

Propriété	Type	Obligatoire	Description
errors	tableau de type chaîne	true	URL d'erreur analysées à partir des balises Error dans la réponse VAST.
vastAds	tableau de type vastAd	true	Publicités analysées à partir de la réponse VAST.
version	chaîne	true	Version de spécification VAST, analysée à partir de l'attribut version de la balise VAST dans la réponse.

vod CreativeOffsets

Cette section décrit les propriétés de l'ADS logs vod. CreativeOffsets

ADS Logs vod Properties CreativeOffsets

Propriété	Type	Obligatoire	Description
<any string>	tableau de type vod CreativeOffset	false	Mappage d'un décalage horaire dans le manifeste avec une liste de publicités à insérer à cet horaire.

vod CreativeOffset

Cette section décrit les propriétés de l'ADS logs vod. CreativeOffset

ADS Logs vod Properties CreativeOffset

Propriété	Type	Obligatoire	Description
adContent	objet de type adContent	true	Informations sur le contenu de la publicité insérée.
creativeUniqueId	chaîne	true	Identifiant unique de la publicité, utilisé comme clé pour le transcodage. Il s'agit du champ ID de la création dans la réponse VAST, si disponible. Sinon, il s'agit de l'URL mezzanine de la publicité.
trackingEvents	objet de type trackingEvents	true	URL de balise de suivi pour les différents

Propriété	Type	Obligatoire	Description
			événements de suivi de la publicité. Les clés sont les noms d'événements et les valeurs sont une liste d'URL de balise.
transcode dAdDuration	number	true	Durée de la publicité , calculée à partir de l'asset transcodé.
uri	chaîne	true	URL de la version mezzanine de la publicité, qui est l'entrée du transcode ur.
vastDuration	number	true	Durée de la publicité , telle qu'analysée à partir de la réponse VAST.

Schéma JSON du journal ADS

Le schéma JSON du journal AWS Elemental MediaTailor ADS est répertorié ci-dessous.

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://amazon.com/elemental/midas/mms/adsLogSchema.json",
  "type": "object",
  "title": "AWS Elemental MediaTailor ADS Log JSON Schema",
  "required": [
    "eventType",
    "eventTimestamp",
    "requestId",
    "sessionType",
    "eventDescription",
```

```

    "awsAccountId",
    "customerId",
    "originId",
    "sessionId"
  ],
  "additionalProperties": false,
  "properties": {
    "eventType": {
      "$id": "#/properties/eventType",
      "type": "string",
      "description": "The code for the event that triggered this log message. Example:
<code>VAST_RESPONSE</code>.",
      "examples": [
        "FILLED_AVAIL"
      ]
    },
    "eventTimestamp": {
      "$id": "#/properties/eventTimestamp",
      "type": "string",
      "description": "The date and time of the event.",
      "examples": [
        "1970-01-01T00:00:00Z"
      ],
      "format": "date-time"
    },
    "requestId": {
      "$id": "#/properties/requestId",
      "type": "string",
      "description": "The MediaTailor request ID, which you can use to correlate
multiple log entries for the same request.",
      "examples": [
        "c7c7ae8c-a61e-44e0-8efd-7723995337a1"
      ],
      "pattern": "^(.*)$"
    },
    "sessionType": {
      "$id": "#/properties/sessionType",
      "type": "string",
      "enum": [
        "HLS",
        "DASH"
      ],
      "description": "The player's stream type."
    }
  },

```



```

    "eventDescription": {
      "$id": "#/properties/eventDescription",
      "type": "string",
      "description": "A short description of the event that triggered this log message,
provided by the MediaTailor service. By default, this is empty. Example: <code>Got
VAST response</code>.",
      "default": "",
      "examples": [
        "Got VAST response"
      ],
      "pattern": "^(.*)$"
    },
    "awsAccountId": {
      "$id": "#/properties/awsAccountId",
      "type": "string",
      "description": "The AWS account ID for the MediaTailor configuration that was
used for the session."
    },
    "customerId": {
      "$id": "#/properties/customerId",
      "type": "string",
      "description": "The hashed version of the AWS account ID, which you can use to
correlate multiple log entries.",
      "pattern": "^(.*)$"
    },
    "originId": {
      "$id": "#/properties/originId",
      "type": "string",
      "description": "The configuration name from the MediaTailor configuration. This
is different from the video content source, which is also part of the configuration.",
      "examples": [
        "external-canary-dash-serverside-reporting-onebox"
      ],
      "pattern": "^(.*)$"
    },
    "sessionId": {
      "$id": "#/properties/sessionId",
      "type": "string",
      "description": "The unique numeric identifier that MediaTailor assigned to the
player session. All requests that a player makes for a session have the same session
ID. Example: <code>e039fd39-09f0-46b2-aca9-9871cc116cde</code>.",
      "examples": [
        "120b9873-c007-40c8-b3db-0f1bd194970b"
      ],
    },
  ],

```

```

    "pattern": "^(.*)$"
  },
  "avail": {
    "$id": "#/properties/avail",
    "type": "object",
    "title": "avail",
    "description": "Information about an avail that MediaTailor fills with ads.
Currently, for the <code>FILLED_AVAIL</code> event type, this is the plan created by
MediaTailor when it first encounters the avail. How the avail is eventually filled may
vary from this plan, depending on how the content plays out. ",
    "required": [
      "creativeAds",
      "originAvailDuration",
      "filledDuration",
      "fillRate",
      "driftMillisecondsAtAvailStart",
      "numAds",
      "slateAd",
      "availId"
    ],
    "additionalProperties": false,
    "properties": {
      "originAvailDuration": {
        "$id": "#/properties/avail/originAvailDuration",
        "type": "number",
        "description": "The duration of the avail as specified in the content stream
from the origin (<code>CUE_OUT</code> or <code>SCTE</code>).",
      },
      "filledDuration": {
        "$id": "#/properties/avail/filledDuration",
        "type": "number",
        "description": "The sum of the durations of all the ads inserted into the
avail."
      },
      "fillRate": {
        "$id": "#/properties/avail/fillRate",
        "type": "number",
        "description": "The rate at which the ads fill the avail duration, from 0.0
(for 0%) to 1.0 (for 100%).",
      },
      "driftMillisecondsAtAvailStart": {
        "$id": "#/properties/avail/driftMillisecondsAtAvailStart",
        "type": "number",

```

```

    "description": "The cumulative drift at the beginning of this avail. A
    positive value implies that we are moving away from the live edge, a negative value
    implies that we are moving towards the live edge."
  },
  "creativeAds": {
    "$id": "#/properties/avail/creativeAds",
    "type": "array",
    "description": "The ads that MediaTailor inserted into the avail.",
    "items": {
      "type": "object",
      "title": "creativeAd",
      "description": "Information about a single inserted ad.",
      "required": [
        "uri",
        "creativeUniqueId",
        "adSystem",
        "adContent",
        "trackingEvents",
        "vastDuration",
        "transcodedAdDuration"
      ],
      "additionalProperties": false,
      "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adSystem": { "$ref": "#/definitions/adSystem" },
        "adContent": { "$ref": "#/definitions/adContent" },
        "trackingEvents": { "$ref": "#/definitions/trackingEvents" },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
      }
    }
  },
  "numAds": {
    "$id": "#/properties/avail/numAds",
    "type": "number",
    "description": "The number of ads that MediaTailor inserted into the avail."
  },
  "slateAd": {
    "$id": "#/properties/avail/slateAd",
    "type": ["object", "null"],
    "title": "slateAd",
    "description": "Information about the slate ad, which MediaTailor uses to
    fill any unfilled segments in the avail.",

```

```

    "additionalProperties": false,
    "required": [
      "uri",
      "creativeUniqueId",
      "adContent",
      "transcodedAdDuration"
    ],
    "properties": {
      "uri": { "$ref": "#/definitions/adMezzanineUri" },
      "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
      "adContent": { "$ref": "#/definitions/adContent" },
      "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
    }
  },
  "availId": {
    "$id": "#/properties/avail/availId",
    "type": "string",
    "description": "The unique identifier for this avail. For HLS, this is the media sequence number where the avail begins. For DASH, this is the period ID."
  },
  "skippedAds": {
    "$id": "#/properties/avail/skippedAds",
    "type": "array",
    "description": "The ads that MediaTailor didn't insert, for reasons like <code>TRANSCODE_IN_PROGRESS</code> and <code>TRANSCODE_ERROR</code>.",
    "items": {
      "type": "object",
      "title": "skippedAd",
      "description": "Information about a single skipped ad.",
      "required": [
        "creativeUniqueId",
        "adMezzanineUrl",
        "skippedReason",
        "vastDuration"
      ],
      "properties": {
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adMezzanineUrl": {
          "type": "string",
          "description": "The mezzanine URL of the skipped ad."
        },
        "skippedReason": {
          "type": "string",

```

```

        "description": "The code that indicates why the ad wasn't inserted.
Example: <code>TRANSCODE_IN_PROGRESS</code>."
    },
    "vastDuration": { "$ref": "#/definitions/vastDuration" },
    "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },
    "targetVariant": {
        "type": "object",
        "title": "targetVariant",
        "description": "The target variant of the source content. This key is
present when an ad wasn't inserted due to the source content containing a variant that
could not match to any variants present in this ad.",
        "required": [
            "mediaProtocol",
            "mediaType",
            "bitrate",
            "mediaResolution",
            "codecs"
        ],
        "additionalProperties": false,
        "properties": {
            "mediaProtocol": {
                "type": "string",
                "description": "The media protocol of this variant, such as HLS.",
                "enum": [
                    "HLS",
                    "DASH"
                ]
            },
            "mediaType": {
                "type": "array",
                "description": "The media type of this variant, such as VIDEO.",
                "items": {
                    "type": "string",
                    "enum": [
                        "VIDEO",
                        "AUDIO",
                        "SUBTITLES",
                        "TRICK_PLAY"
                    ]
                },
                "description": "Media type, such as VIDEO."
            }
        }
    },
    "bitrate": {
        "$ref": "#/definitions/bitrate"
    }

```

```

    },
    "mediaResolution": {
      "type": "object",
      "title": "mediaResolution",
      "description": "The media resolution of this variant.",
      "required": [
        "width",
        "height"
      ],
      "additionalProperties": false,
      "properties": {
        "width": {
          "$ref": "#/definitions/width"
        },
        "height": {
          "$ref": "#/definitions/height"
        }
      }
    },
    "codecs": {
      "type": "array",
      "description": "The codecs of this variant.",
      "items": {
        "type": "string",
        "description": "Codec, such as avc1."
      }
    }
  }
}
},
"adBreakTrackingEvents": {
  "$id": "#properties/avail/adBreakTrackingEvents",
  "type": "object",
  "title": "adBreakTrackingEvents",
  "description": "VMAP/ad break tracking events and corresponding URL",
  "properties": {
    "items": {
      "$ref": "#/definitions/adBreakTrackingEvents"
    }
  }
}
}
}

```

```

    },

    "vastResponse": {
      "$id": "#/properties/vastResponse",
      "type": "object",
      "title": "vastResponse",
      "description": "Information about the VAST response that MediaTailor received
from the ADS.",
      "required": [
        "version",
        "vastAds",
        "errors",
        "nonLinearAdsList"
      ],
      "additionalProperties": false,
      "properties": {
        "version": {
          "$id": "#/properties/vastResponse/version",
          "type": "string",
          "description": "The VAST specification version, parsed from the
<code>version</code> attribute of the <code>VAST</code> tag in the response.",
          "examples": [
            "3.0"
          ],
          "pattern": "^(.*)$"
        },
        "vastAds": {
          "$id": "#/properties/vastResponse/vastAds",
          "type": "array",
          "description": "The ads parsed from the VAST response.",
          "items": {
            "$ref": "#/definitions/vastAd"
          }
        },
        "errors": {
          "$id": "#/properties/vastResponse/errors",
          "type": "array",
          "description": "The error URLs parsed from the <code>Error</code> tags in the
VAST response.",
          "items": {
            "type": "string",
            "description": "A single error URL."
          }
        }
      }
    },
  },

```

```

    "nonLinearAdsList": {
      "$id": "#/properties/vastResponse/nonLinearAds",
      "type": "array",
      "description": "A list of NonLinearAds as they are read from the VAST
response.",
      "items": {
        "$ref": "#/definitions/nonLinearAds"
      }
    }
  },

  "vastAd": {
    "$ref": "#/definitions/vastAd"
  },

  "vodVastResponseTimeOffset": {
    "$id": "#/properties/vodVastResponseTimeOffset",
    "type": "number",
    "description": "The VMAP specific time offset for VOD ad insertion.",
    "examples": [
      5.0
    ]
  },

  "vodCreativeOffsets": {
    "$id": "#/properties/vodCreativeOffsets",
    "type": "object",
    "title": "vodCreativeOffsets",
    "description": "A map that indicates the time offsets in the manifest where
MediaTailor will insert avails, based on the VMAP response.",
    "additionalProperties": {
      "type": "array",
      "$id": "#/properties/vodCreativeOffsets/entry",
      "description": "A mapping from a time offset in the manifest to a list of ads
to insert at this time.",
      "items": {
        "type": "object",
        "$id": "#/properties/vodCreativeOffsets/entry/items",
        "title": "vodCreativeOffset",
        "description": "The list of ads to insert at the specified time offset.",
        "additionalProperties": false,
        "required": [
          "uri",

```



```

        "creativeUniqueId",
        "vastDuration",
        "transcodedAdDuration",
        "adContent",
        "trackingEvents"
    ],
    "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },
        "adContent": { "$ref": "#/definitions/adContent" },
        "trackingEvents": { "$ref": "#/definitions/trackingEvents" }
    }
}
},
"adsRequestUrl": {
    "$id": "#/properties/adsRequestUrl",
    "type": "string",
    "description": "The full URL of the ADS request made by MediaTailor."
},
"adMarkers": {
    "$id": "#/properties/adMarkers",
    "type": "string",
    "description": "Found Ad Marker in the Manifest."
},
"segmentationUpid": {
    "$id": "#/properties/segmentationUpid",
    "type": "string",
    "description": "Value of segmentation upid parsed from ad markers in manifest."
},
"segmentationTypeId": {
    "$id": "#/properties/segmentationTypeId",
    "type": "integer",
    "description": "Value of segmentation typeId parsed from ad markers in manifest."
},
"requestHeaders": {
    "$id": "#/properties/requestHeaders",
    "type": "array",
    "description": "The headers that MediaTailor included with the ADS request.
Typically, the logs include these when a request to the ADS fails, to help with
troubleshooting.",

```

```

    "items": {
      "type": "object",
      "title": "requestheaders",
      "description": "The name and value for a single header included in the ADS
request.",
      "required": [
        "name",
        "value"
      ],
      "additionalProperties": false,
      "properties": {
        "name": {
          "type": "string",
          "description": "The name of the header."
        },
        "value": {
          "type": "string",
          "description": "The value of the header."
        }
      }
    }
  },

  "originalTargetUrl": {
    "$id": "#/properties/originalTargetUrl",
    "type": "string",
    "description": "The old URL to which MediaTailor was going to make a request."
  },

  "prefetchScheduleName": {
    "$id": "#/properties/prefetchScheduleName",
    "type": "string",
    "description": "The name of the prefetch schedule associated with this ad
event.",
    "examples": [
      "PrefetchScheduleName"
    ],
    "pattern": "^(.*)$"
  },

  "updatedTargetUrl": {
    "$id": "#/properties/updatedTargetUrl",
    "type": "string",
    "description": "The new URL to which MediaTailor is making a request."
  },

```

```
"rawAdsResponse": {
  "$id": "#/properties/rawAdsResponse",
  "type": "string",
  "description": "Paginated ADS response as it's exactly returned to MediaTailor."
},
"rawAdsResponseIndex": {
  "$id": "#/properties/rawAdsResponseIndex",
  "type": "integer",
  "description": "Integer value denoting this rawAdsResponse's index into the
full ADS response. This value is used to order the paginated messages for this ADS
response."
},

"eventId": {
  "$id": "#/properties/eventId",
  "type": "string",
  "description": "Unique identifier per hook invocation. Use this field to
correlate hook-level and function-level events for the same execution.",
  "examples": [
    "5dc6f040-0f72-4e8c-a64e-25eeef62708c"
  ]
},
"functionId": {
  "$id": "#/properties/functionId",
  "type": "string",
  "description": "The identifier of the function. On hook events, this is the
function configured on the hook. On function events, this is the individual function
within the executor.",
  "examples": [
    "my_first_function"
  ]
},
"functionType": {
  "$id": "#/properties/functionType",
  "type": "string",
  "enum": [
    "CUSTOM_OUTPUT",
    "HTTP_REQUEST",
    "SEQUENTIAL_EXECUTOR",
    "CONCURRENT_EXECUTOR"
  ],
  "description": "The type of function."
},
"executionTimeMs": {
```

```

    "$id": "#/properties/executionTimeMs",
    "type": "integer",
    "description": "Execution time in milliseconds."
  },
  "status": {
    "$id": "#/properties/status",
    "type": "string",
    "enum": [
      "SUCCESS",
      "ERROR"
    ],
    "description": "Hook execution outcome. Present on HOOK_SUMMARY events only."
  },
  "errorType": {
    "$id": "#/properties/errorType",
    "type": "string",
    "enum": [
      "INTERNAL_ERROR",
      "SYNTAX_ERROR",
      "RESOURCE_LIMIT_ERROR",
      "RESTRICTION_ERROR",
      "TIMEOUT_ERROR",
      "VALIDATION_ERROR"
    ],
    "description": "Structured error classification. Present on error events only."
  },
  "cause": {
    "$id": "#/properties/cause",
    "type": "string",
    "description": "Error detail string describing the specific failure."
  },
  "input": {
    "$id": "#/properties/input",
    "type": "object",
    "description": "Before-values of variables targeted by output expressions."
  },
  "output": {
    "$id": "#/properties/output",
    "type": "object",
    "description": "Values produced by the function. Present on FUNCTION_COMPLETED events only."
  },
  "http": {
    "$id": "#/properties/http",

```

```
"type": "object",
"description": "HTTP request and response details for HTTP_REQUEST functions.",
"properties": {
  "request": {
    "type": "object",
    "description": "The HTTP request details.",
    "properties": {
      "url": { "type": "string", "description": "The request URL." },
      "method": { "type": "string", "description": "The HTTP method." },
      "headers": {
        "type": "array",
        "description": "The request headers.",
        "items": {
          "type": "object",
          "properties": {
            "name": { "type": "string" },
            "value": { "type": "string" }
          }
        }
      },
      "body": { "type": ["string", "null"], "description": "The request body." }
    },
    "responseTimeMs": { "type": ["integer", "null"], "description": "HTTP response time in milliseconds." },
    "response": {
      "type": ["object", "null"],
      "description": "The HTTP response details. Null when the server was never reached.",
      "properties": {
        "statusCode": { "type": ["integer", "null"], "description": "The HTTP status code. Null when the server was never reached." },
        "headers": {
          "type": ["array", "null"],
          "description": "The response headers.",
          "items": {
            "type": "object",
            "properties": {
              "name": { "type": "string" },
              "value": { "type": "string" }
            }
          }
        },
        "body": { "type": ["string", "null"], "description": "The response body." }
```

```

    }
  },
  "truncated": { "type": "boolean", "description": "True when request or response
body was truncated to fit within log event size limits." }
}
}
},

```

"__COMMENT_oneOf": "The oneOf section defines subtypes for our events. Subtypes can have different rules, including which fields are required. For more information, see <https://json-schema.org/understanding-json-schema/reference/combining.html#oneof> ",

```

"oneOf": [
  { "$ref": "#/definitions/eventAdMarkersFound" },
  { "$ref": "#/definitions/eventNonAdMarkerFound" },
  { "$ref": "#/definitions/eventMakingAdsRequest" },
  { "$ref": "#/definitions/eventModifiedTargetUrl" },
  { "$ref": "#/definitions/eventRawAdsResponse" },
  { "$ref": "#/definitions/eventVastResponse" },
  { "$ref": "#/definitions/eventFilledAvail" },
  { "$ref": "#/definitions/eventFilledOverlayAvail" },
  { "$ref": "#/definitions/eventErrorFiringBeaconFailed" },
  { "$ref": "#/definitions/eventWarningNoAdvertisements" },
  { "$ref": "#/definitions/eventUnknownHost" },
  { "$ref": "#/definitions/eventErrorAdsTimeout" },
  { "$ref": "#/definitions/eventErrorVastMissingOverlays" },
  { "$ref": "#/definitions/eventPlannedAvail" },
  { "$ref": "#/definitions/eventEmptyVastResponse" },
  { "$ref": "#/definitions/eventEmptyVmapResponse" },
  { "$ref": "#/definitions/eventErrorUnknown" },
  { "$ref": "#/definitions/eventVastRedirect" },
  { "$ref": "#/definitions/eventRedirectedVastResponse" },
  { "$ref": "#/definitions/eventErrorAdsMissingImpression" },
  { "$ref": "#/definitions/eventErrorAdsResponseParse" },
  { "$ref": "#/definitions/eventErrorAdsInvalidResponse" },
  { "$ref": "#/definitions/eventErrorDisallowedHost" },
  { "$ref": "#/definitions/eventPersonalizationDisabled" },
  { "$ref": "#/definitions/eventWarningDynamicVariableSubFailed" },
  { "$ref": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset" },
  { "$ref": "#/definitions/eventVodTimeBasedAvailPlanSuccess" },
  { "$ref": "#/definitions/eventPreAdsRequestHookSummary" },
  { "$ref": "#/definitions/eventPreAdsRequestHookError" },
  { "$ref": "#/definitions/eventPreAdsRequestFunctionCompleted" },
  { "$ref": "#/definitions/eventPreAdsRequestFunctionError" }
]

```

```
],  
  
"definitions": {  
  "eventAdMarkersFound": {  
    "$id": "#/definitions/eventAdMarkersFound",  
    "required": [  
      "eventType",  
      "adMarkers"  
    ],  
    "properties": {  
      "eventType": {  
        "type": "string",  
        "const": "AD_MARKER_FOUND"  
      }  
    }  
  },  
  "eventNonAdMarkerFound": {  
    "$id": "#/definitions/eventNonAdMarkerFound",  
    "required": [  
      "eventType",  
      "adMarkers"  
    ],  
    "properties": {  
      "eventType": {  
        "type": "string",  
        "const": "NON_AD_MARKER_FOUND"  
      }  
    }  
  },  
  "eventMakingAdsRequest": {  
    "$id": "#/definitions/eventMakingAdsRequest",  
    "required": [  
      "eventType",  
      "adsRequestUrl"  
    ],  
    "properties": {  
      "eventType": {  
        "type": "string",  
        "const": "MAKING_ADS_REQUEST"  
      }  
    }  
  },  
  "eventModifiedTargetUrl": {
```

```

    "$id": "#/definitions/eventModifiedTargetUrl",
    "required": [
        "eventType",
        "originalTargetUrl",
        "updatedTargetUrl"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "MODIFIED_TARGET_URL"
        }
    }
},

```

```

"eventRawAdsResponse": {
    "$id": "#/definitions/eventRawAdsResponse",
    "required": [
        "eventType",
        "rawAdsResponse",
        "rawAdsResponseIndex"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "RAW_ADS_RESPONSE"
        }
    }
},

```

```

"eventVastResponse": {
    "$id": "#/definitions/eventVastResponse",
    "_comment": "NOTE: the vastResponse property should ideally be marked as a
required field for this event, but unfortunately, in the case of an empty vast
response, we currently emit an EMPTY_VAST_RESPONSE followed by a VAST_RESPONSE, and
the vastResponse property is not present in the latter. We need to fix this so that we
don't emit both of those events in the empty response case, and update this schema to
flag vastResponse as required for VAST_RESPONSE.",
    "required": [
        "eventType"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "VAST_RESPONSE"
        }
    }
}

```



```

    }
  }
},

"eventFilledAvail": {
  "$id": "#/definitions/eventFilledAvail",
  "required": [
    "eventType",
    "avail"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "FILLED_AVAIL"
    }
  }
},

"eventFilledOverlayAvail": {
  "$id": "#/definitions/eventFilledOverlayAvail",
  "required": [
    "eventType",
    "avail"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "FILLED_OVERLAY_AVAIL"
    }
  }
},

"eventErrorVastMissingOverlays": {
  "$id": "#/definitions/eventErrorVastMissingOverlays",
  "required": [
    "eventType",
    "adsRequestUrl",
    "requestHeaders"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_VAST_MISSING_OVERLAYS"
    }
  }
}

```

```
    }
  },

  "eventErrorFiringBeaconFailed": {
    "$id": "#/definitions/eventErrorFiringBeaconFailed",
    "required": [
      "eventType",
      "error",
      "beaconInfo"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_FIRING_BEACON_FAILED"
      }
    }
  },

  "eventWarningNoAdvertisements": {
    "$id": "#/definitions/eventWarningNoAdvertisements",
    "required": [
      "eventType"
    ],
    "_comment": "We should really have a more descriptive error field for these events",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "WARNING_NO_ADVERTISEMENTS"
      }
    }
  },

  "eventUnknownHost": {
    "$id": "#/definitions/eventUnknownHost",
    "required": [
      "eventType",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_UNKNOWN_HOST"
      }
    }
  }
}
```

```
    }
  },

  "eventErrorAdsTimeout": {
    "$id": "#/definitions/eventErrorAdsTimeout",
    "required": [
      "eventType",
      "adsRequestUrl",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_TIMEOUT"
      }
    }
  },

  "eventPlannedAvail": {
    "$id": "#/definitions/eventPlannedAvail",
    "required": [
      "eventType"
    ],
    "_comment": "TODO: Flesh this out as we implement it",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PLANNED_AVAIL"
      }
    }
  },

  "eventEmptyVastResponse": {
    "$id": "#/definitions/eventEmptyVastResponse",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "EMPTY_VAST_RESPONSE"
      }
    }
  },
}
```

```
"eventEmptyVmapResponse": {
  "$id": "#/definitions/eventEmptyVmapResponse",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "EMPTY_VMAP_RESPONSE"
    }
  }
},

"eventErrorUnknown": {
  "$id": "#/definitions/eventErrorUnknown",
  "required": [
    "eventType"
  ],
  "_comment": "TODO: we should have a field for the exception message or something",
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_UNKNOWN"
    }
  }
},

"eventVastRedirect": {
  "$id": "#/definitions/eventVastRedirect",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "VAST_REDIRECT"
    }
  }
},

"eventRedirectedVastResponse": {
  "$id": "#/definitions/eventRedirectedVastResponse",
```

```
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "REDIRECTED_VAST_RESPONSE"
      }
    },
    "_comment": "NOTE that the property vastResponse is not required because empty vast responses do not contain a vastResponse."
  },

  "eventErrorAdsResponseParse": {
    "$id": "#/definitions/eventErrorAdsResponseParse",
    "required": [
      "eventType"
    ],
    "_comment": "We should have a field with an error message here",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_RESPONSE_PARSE"
      }
    }
  },

  "eventErrorAdsInvalidResponse": {
    "$id": "#/definitions/eventErrorAdsInvalidResponse",
    "required": [
      "eventType",
      "additionalInfo"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_INVALID_RESPONSE"
      }
    }
  },

  "eventErrorAdsMissingImpression": {
    "$id": "#/definitions/eventErrorAdsMissingImpression",
    "required": [
```

```
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_VAST_MISSING_IMPRESSION"
    }
  }
},

"eventErrorDisallowedHost": {
  "$id": "#/definitions/eventErrorDisallowedHost",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_DISALLOWED_HOST"
    }
  }
},

"eventPersonalizationDisabled": {
  "$id": "#/definitions/eventPersonalizationDisabled",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_PERSONALIZATION_DISABLED"
    }
  }
},

"eventWarningDynamicVariableSubFailed": {
  "$id": "#/definitions/eventWarningDynamicVariableSubFailed",
  "required": [
    "eventType",
    "adsRequestUrl"
  ],
  "properties": {
    "eventType": {
```

```

        "type": "string",
        "const": "WARNING_URL_VARIABLE_SUBSTITUTION_FAILED"
    }
}
},

"eventVodTimeBasedAvailPlanVastResponseForOffset": {
    "$id": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset",
    "required": [
        "eventType",
        "vastResponse"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET"
        }
    }
},

"eventVodTimeBasedAvailPlanSuccess": {
    "$id": "#/definitions/eventVodTimeBasedAvailPlanSuccess",
    "required": [
        "eventType",
        "vodCreativeOffsets"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "VOD_TIME_BASED_AVAIL_PLAN_SUCCESS"
        }
    }
},

"creativeUniqueId": {
    "type": "string",
    "description": "The unique identifier for the ad, used as a key for transcoding.
This is the ID field for the creative in the VAST response, if available. Otherwise,
it's the mezzanine URL of the ad. "
},

"adSystem": {
    "type": "string",

```

```

    "description": "The value of the <code>AdSystem</code> tag in the VAST response.
"
  },

  "vastDuration": {
    "type": "number",
    "description": "The duration of the ad, as parsed from the VAST response."
  },

  "transcodedAdDuration": {
    "type": "number",
    "description": "The duration of the ad, calculated from the transcoded asset."
  },

  "adContent": {
    "$id": "#/properties/adContent",
    "type": ["object", "null"],
    "title": "adContent",
    "description": "Information about the content of the inserted ad.",
    "additionalProperties": false,
    "properties": {
      "adPlaylistUris": {
        "$id": "#/properties/adContent/adPlaylistUris",
        "type": "object",
        "title": "adPlaylistUris",
        "description": "The mapping from the origin manifest for a variant to the ad
manifest for the variant. For DASH, this contains a single entry, because all variants
are represented in a single DASH manifest. ",
        "additionalProperties": {
          "$id": "#/properties/adContent/adPlaylistUris/adPlaylistUri",
          "type": "string",
          "description": "The URL of the ad manifest for the specific variant."
        }
      }
    }
  },

  "adMezzanineUri": {
    "type": "string",
    "description": "The URL of the mezzanine version of the ad, which is the input to
the transcoder."
  },

  "bitrate": {

```



```
    "type": "integer",
    "examples": [
      533
    ],
    "description": "The bitrate."
  },
  "width": {
    "type": "integer",
    "examples": [
      1280
    ],
    "description": "Width in pixels."
  },
  "height": {
    "type": "integer",
    "examples": [
      720
    ],
    "description": "Height in pixels."
  },
  "trackingEvents": {
    "type": "object",
    "title": "trackingEvents",
    "description": "The tracking beacon URLs for the various tracking events for the
ad. The keys are the event names, and the values are a list of beacon URLs.",
    "additionalProperties": {
      "type": "array",
      "description": "The list of beacon URLs for the specified tracking event
(impression, complete, and so on)",
      "items": {
        "type": "string",
        "description": "The beacon URLs for this tracking event."
      }
    }
  },
  "nonLinearAds": {
    "$id": "#/properties/nonLinearAds",
    "type": "object",
    "title": "nonLinearAds",
    "description": "A NonLinearAds as it appears in the VAST response.",
    "required": [
```

```

        "nonLinearAdList",
        "nonLinearTrackingEvents"
    ],
    "properties": {
        "nonLinearAdList": {
            "type": "array",
            "description": "List of non linear ads as they exist within one
NonLinearAds.",
            "items": {
                "type": "object",
                "title": "nonLinearAdList",
                "description": "List of NonLinearAd as they are parsed from its parent
NonLinearAds.",
                "properties": {
                    "nonLinearAdId": {
                        "type": "string",
                        "description": "Ad ID of this non linear ad."
                    },
                    "nonLinearAdSystem": {
                        "type": "string",
                        "description": "Ad system of this non linear ad's parent Inline ad."
                    },
                    "nonLinearAdTitle": {
                        "type": "string",
                        "description": "Ad title of this non linear ad's parent Inline ad."
                    },
                    "nonLinearCreativeId": {
                        "type": "string",
                        "description": "Creative ID of this non linear ad's parent Creative
ad."
                    },
                    "nonLinearCreativeAdId": {
                        "type": "string",
                        "description": "Creative ad ID of this non linear ad."
                    },
                    "nonLinearCreativeSequence": {
                        "type": "string",
                        "description": "Creative sequence of this non linear ad."
                    },
                    "nonLinearWidth": {
                        "type": "string",
                        "description": "Width of this non linear ad."
                    },
                    "nonLinearHeight": {

```

```

    "type": "string",
    "description": "Height of this non linear ad."
  },
  "nonLinearExpandedWidth": {
    "type": "string",
    "description": "Expanded width of this non linear ad."
  },
  "nonLinearExpandedHeight": {
    "type": "string",
    "description": "Expanded height of this non linear ad."
  },
  "nonLinearScalable": {
    "type": "boolean",
    "description": "Boolean denoting if this non linear ad is scalable."
  },
  "nonLinearMaintainAspectRatio": {
    "type": "boolean",
    "description": "Boolean denoting if aspect ratio should be maintained
for this non linear ad."
  },
  "nonLinearMinSuggestedDuration": {
    "type": "string",
    "description": "Min suggested duration for this non linear ad."
  },
  "nonLinearApiFramework": {
    "type": "string",
    "description": "API framework for this non linear ad's parent Inline
ad."
  },
  "nonLinearStaticResource": {
    "type": "string",
    "description": "Static resource for this non linear ad."
  },
  "nonLinearStaticResourceCreativeType": {
    "type": "string",
    "description": "Static Resource creative type for this non linear ad."
  },
  "nonLinearIframeResource": {
    "type": "string",
    "description": "I-Frame resource for this non linear ad."
  },
  "nonLinearHtmlResource": {
    "type": "string",
    "description": "HTML resource for this non linear ad."
  }

```

```

    },
    "nonLinearAdParameters": {
      "type": "string",
      "description": "Ad parameters for this non linear ad."
    },
    "nonLinearClickThrough": {
      "type": "string",
      "description": "Click Through data for this non linear ad."
    },
    "nonLinearClickTracking": {
      "type": "string",
      "description": "Click Tracking data for this non linear ad."
    },
    "nonLinearClickTrackingId": {
      "type": "string",
      "description": "Click Tracking ID for this non linear ad."
    }
  }
},
"nonLinearTrackingEvents": { "$ref": "#/definitions/trackingEvents" },
"extensions": {
  "$id": "#/properties/nonLinearAds/extensions",
  "type": "array",
  "description": "Extensions that exist for this NonLinearAds.",
  "items": {
    "$id": "#/properties/nonLinearAds/extensions/items",
    "type": "object",
    "title": "Extensions",
    "description": "Extensions found in non linear ads",
    "additionalProperties": false,
    "properties": {
      "extensionType": {
        "$id": "#/properties/nonLinearAds/extensions/extensionType",
        "type": "string",
        "description": "The value of the extension type attribute of the
<code>Extensions</code> tag.",
        "examples": [
          "FreeWheel"
        ]
      },
      "extensionContent": {
        "$id": "#/properties/nonLinearAds/extensions/extensionContent",
        "type": "string",

```

```

        "description": "The extension content attribute of the
<code>Extensions</code> tag.",
        "examples": [
            "progressive"
        ]
    }
}
}
}
},
"adBreakTrackingEvents": {
    "$id": "#/properties/adBreakTrackingEvents",
    "type": "object",
    "title": "adBreakTrackingEvents",
    "description": "These are all VMAP ad break tracking events.",
    "additionalProperties": {
        "type": "array",
        "description": "VMAP/ad break tracking events and corresponding URL",
        "items": {
            "type": "string",
            "description": "The beacon URLs for this tracking event."
        }
    }
},
"vastAd": {
    "$id": "#/properties/vastAd",
    "type": "object",
    "title": "vastAd",
    "description": "Information about a single ad parsed from the VAST response.",
    "required": [
        "vastAdId",
        "adSystem",
        "adTitle",
        "creativeId",
        "creativeAdId",
        "duration",
        "vastMediaFiles",
        "trackingEvents"
    ],
    "additionalProperties": false,
    "properties": {
        "vastAdId": {
            "$id": "#/properties/vastAd/vastAdId",

```

```

        "type": "string",
        "description": "The value of the id attribute of the <code>Ad</code> tag in
the VAST response",
        "examples": [
            "ad1"
        ]
    },
    "adSystem": {"$ref": "#/definitions/adSystem" } ,
    "adTitle": {
        "$id": "#/properties/vastAd/adTitle",
        "type": "string",
        "description": "The media files that are available for the ad in the VAST
response.",
        "examples": [
            "External NCA1C1L1 LinearInlineSkippable"
        ]
    },
    "creativeId": {
        "$id": "#/properties/vastAd/creativeId",
        "type": "string",
        "description": "The value of the id attribute of the <code>Creative</code>
tag in the VAST response.",
        "examples": [
            "creative1"
        ]
    },
    "creativeAdId": {
        "$id": "#/properties/vastAd/creativeAdId",
        "type": "string",
        "description": "The value of the adId attribute of the <code>Creative</code>
tag in the VAST response."
    },
    "duration": {
        "$id": "#/properties/vastAd/duration",
        "type": "number",
        "description": "The approximate duration of the ad, based on the
<code>duration</code> tag in the <code>linear</code> element of the VAST response.",
        "examples": [
            30,
            30.0
        ]
    },
    "vastMediaFiles": {
        "$id": "#/properties/vastAd/vastMediaFiles",

```

```

    "type": "array",
    "description": "The list of available media files for the ad in the VAST
response.",
    "items": {
      "$id": "#/properties/vastAd/vastMediaFiles/items",
      "type": "object",
      "title": "vastMediaFile",
      "description": "Information about a media file for the ad.",
      "required": [
        "uri",
        "id",
        "delivery",
        "type",
        "apiFramework",
        "width",
        "height",
        "bitrate"
      ],
      "additionalProperties": false,
      "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "id": {
          "$id": "#/properties/vastAd/vastMediaFiles/items/properties/id",
          "type": "string",
          "description": "The value of the id attribute of the <code>MediaFile</
code> tag.",
          "examples": [
            "GDFP"
          ]
        },
        "delivery": {
          "$id": "#/properties/vastAd/vastMediaFiles/items/properties/delivery",
          "type": "string",
          "description": "The protocol used for the media file, set to either
progressive or streaming.",
          "examples": [
            "progressive"
          ]
        },
        "type": {
          "$id": "#/properties/vastAd/vastMediaFiles/items/properties/type",
          "type": "string",
          "description": "The MIME type of the media file, taken from the type
attribute of the <code>MediaFile</code> tag.",

```

```

        "examples": [
            "video/mp4"
        ],
    },
    "apiFramework": {
        "$id": "#/properties/vastAd/vastMediaFiles/items/properties/
apiFramework",
        "type": "string",
        "description": "The API framework needed to manage the media file.
Example: <code>VPAID</code>."
    },
    "width": {
        "$ref": "#/definitions/width"
    },
    "height": {
        "$ref": "#/definitions/height"
    },
    "bitrate": {
        "$ref": "#/definitions/bitrate"
    }
}
},
"trackingEvents": { "$ref": "#/definitions/trackingEvents" },
"vastAdTagUri": {
    "$id": "#/properties/vastAd/vastAdTagUri",
    "type": "string",
    "description": "The VMAP-specific redirect URI for an ad.",
    "examples": [
        "https://ads.redirect.com/redirect1"
    ]
}
},
"eventPreAdsRequestHookSummary": {
    "$id": "#/definitions/eventPreAdsRequestHookSummary",
    "required": [
        "eventType",
        "eventId",
        "functionId",
        "executionTimeMs",
        "status"
    ],

```



```
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PRE_ADS_REQUEST_HOOK_SUMMARY"
      }
    }
  },

  "eventPreAdsRequestHookError": {
    "$id": "#/definitions/eventPreAdsRequestHookError",
    "required": [
      "eventType",
      "eventId",
      "functionId",
      "executionTimeMs",
      "errorType",
      "cause"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PRE_ADS_REQUEST_HOOK_ERROR"
      }
    }
  },

  "eventPreAdsRequestFunctionCompleted": {
    "$id": "#/definitions/eventPreAdsRequestFunctionCompleted",
    "required": [
      "eventType",
      "eventId",
      "functionId",
      "functionType",
      "executionTimeMs"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PRE_ADS_REQUEST_FUNCTION_COMPLETED"
      }
    }
  },

  "eventPreAdsRequestFunctionError": {
```

```
    "$id": "#/definitions/eventPreAdsRequestFunctionError",
    "required": [
      "eventType",
      "eventId",
      "functionId",
      "functionType",
      "executionTimeMs",
      "errorType",
      "cause"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PRE_ADS_REQUEST_FUNCTION_ERROR"
      }
    }
  }
}
```

AWS Elemental MediaTailor description des journaux manifestes et types d'événements

Les sections suivantes décrivent les journaux qui MediaTailor sont émis pour décrire les événements survenus avec le serveur d'origine lors de la demande et de la réception d'un manifeste. Ce sont ManifestService des journaux.

Rubriques

- [ManifestService events](#)
- [Propriétés des journaux du manifeste](#)

ManifestService events

Les événements suivants sont émis lors MediaTailor des interactions avec l'origine.

Journal	Description
CONFIG_SECURITY_ERROR	La MediaTailor configuration présente un problème de sécurité.
CONFIG_SYNTAX_ERROR	L'origine et le chemin de l'actif entraînent une URL mal formée.
CONNECTION_ERROR	La MediaTailor connexion à l'origine a été refusée ou a échoué.
GENERATED_MANIFEST	MediaTailor a généré un manifeste. Le mode débogage doit être activé pour recevoir ces journaux. Pour plus d'informations sur le mode journal de débogage, notamment sur la manière de l'activer, consultez Génération de journaux de débogage.
HOST_DISALLOWED	MediaTailor n'autorise pas les requêtes HTTP vers cet hôte.
INCOMPATIBLE_HLS_VERSION	Le manifeste utilise une version HLS incompatible. MediaTailor nécessite la version 3 ou supérieure.
INVALID_SINGLE_PERIOD_DASH_MANIFEST	Le manifeste DASH à période unique n'est pas valide. MediaTailor passe par un manifeste DASH à période unique.
IO_ERROR	MediaTailor J'ai rencontré une erreur d'E/S lors de la communication avec l'origine.
LAST_PERIOD_MISSING_AUDIO	La dernière période du manifeste DASH ne contient aucun son en AdaptationSets raison d'un mauvais alignement audio ou vidéo d'origine. Pour éviter les problèmes de lecture, retardez la publication de la dernière période au moins jusqu'à la prochaine demande.

Journal	Description
LAST_PERIOD_MISSING_AUDIO_WARNING	La dernière période du manifeste DASH ne contient aucun son en AdaptationSets raison d'un mauvais alignement audio ou vidéo d'origine. Choisir de publier (sans retarder) la dernière période. L'absence de son peut entraîner des problèmes de lecture.
MANIFEST_ERROR	La demande de MediaTailor manifeste a échoué.
NO_MASTER_OR_MEDIA_PLAYLIST	La réponse d'origine ne contient pas de playlist principale ou de playlist multimédia.
NO_MASTER_PLAYLIST	La réponse d'origine ne contient pas la liste de lecture principale attendue.
NO_MEDIA_PLAYLIST	La réponse d'origine ne contient pas la playlist multimédia attendue.
ORIGIN_MANIFEST	MediaTailor a récupéré un manifeste d'origine. Le mode débogage doit être activé pour recevoir ces journaux. Pour plus d'informations sur le mode journal de débogage, notamment sur la manière de l'activer, consultez Génération de journaux de débogage .
PARSING_ERROR	L'origine n'est pas en mesure d'analyser la demande de manifeste.
PRE_SESSION_INIT_FUNCTION_COMPLETED	Une fonction individuelle du hook d'initialisation de pré-session est terminée. Il s'agit d'un type d'événement optionnel.
PRE_SESSION_INIT_FUNCTION_ERROR	Une fonction individuelle du hook d'initialisation de pré-session a échoué.

Journal	Description
PRE_SESSION_INIT_HOOK_ERROR	L'exécution du hook d'initialisation de pré-session a échoué.
PRE_SESSION_INIT_HOOK_SUMMARY	Résumé de l'exécution du hook d'initialisation avant la session, y compris l'état de réussite ou d'erreur. Il s'agit d'un type d'événement optionnel.
SCTE35_PARSING_ERROR	MediaTailor est incapable d'analyser l'Signal Binary élément du manifeste.
SESSION_INITIALIZED	Une session a été initialisée. Le mode débogage doit être activé pour recevoir ces journaux. Pour plus d'informations sur le mode journal de débogage, notamment sur la manière de l'activer, consultez Génération de journaux de débogage.
TIMEOUT_ERROR	Le délai de la demande de MediaTailor manifeste a expiré.
TRACKING_RESPONSE	MediaTailor a envoyé une réponse de suivi. Le mode débogage doit être activé pour recevoir ces journaux. Pour plus d'informations sur le mode journal de débogage, notamment sur la manière de l'activer, consultez Génération de journaux de débogage.
UNKNOWN_ERROR	MediaTailor a rencontré une erreur inconnue.
UNKNOWN_HOST	L'hôte est inconnu.
UNSUPPORTED_SINGLE_PERIOD_DASH_MANIFEST	Le manifeste DASH à période unique n'est pas pris en charge. MediaTailor passe par un manifeste DASH à période unique.

Propriétés des journaux du manifeste

Cette section décrit les propriétés des journaux du manifeste.

Propriété	Type	Obligatoire
awsAccountId	chaîne	true
eventTimestamp	chaîne	true
originId	chaîne	true
customerId	chaîne	false
eventType	chaîne	false
sessionId	chaîne	false
originRequestUrl	chaîne	false
mediaTailorPath	chaîne	false
requestId	chaîne	false
responseBody	chaîne	false
sessionType	string (valeurs légales : [DASH, HLS])	false
requestNextToken	chaîne	false
eventDescription	chaîne	false
errorType	chaîne	false
eventId	chaîne	false
executionTimeMs	entier	false
functionId	chaîne	false
functionType	chaîne	false

Propriété	Type	Obligatoire
http	objet	false
input	objet	false
output	objet	false
status	chaîne	false
assetPath	chaîne	false
originFullUrl	chaîne	false
originPrefixUrl	chaîne	false
additionalInfo	chaîne	false
cause	chaîne	false
response	chaîne	false
httpCode	chaîne	false
errorMessage	chaîne	false
adAdsResponse	chaîne	false
adAdsRawResponse	chaîne	false
adAdsRequest	chaîne	false
adNumNewAvails	chaîne	false
generatedMediaPlay list	chaîne	false
requestStartTime	chaîne	false
requestEndTime	chaîne	false

Propriété	Type	Obligatoire
<code>requestStartTimeEpochMillis</code>	chaîne	false
<code>requestEndTimeEpochMillis</code>	chaîne	false

AWS Elemental MediaTailor description des journaux de transcodage et types d'événements

Les sections suivantes décrivent les journaux qui sont émis pour décrire les événements liés au service de transcodage lors de la préparation de créations pour l' MediaTailor assemblage d'annonces. Ce sont TranscodeService des journaux.

Rubriques

- [TranscodeService events](#)
- [Propriétés des journaux de transcodage](#)

TranscodeService events

Les événements suivants sont émis lors des MediaTailor interactions lors du transcodage des publicités.

Journal	Description
IMPORT_ERROR	MediaTailor J'ai rencontré une erreur interne lors d'une tâche d'importation (pour les annonces préconditionnées). En utilisant un ensemble d'annonces vide.
INITIALIZED	MediaTailor a initialisé soit une tâche de transcodage, soit une tâche d'importation (pour les publicités préconditionnées).

Journal	Description
INTERNAL_ERROR	MediaTailor a rencontré une erreur interne. En utilisant un ensemble d'annonces vide.
MISSING_VARIANTS	MediaTailor Impossible de transcoder l'annonce en raison de variantes manquantes. En utilisant un ensemble d'annonces vide.
PROFILE_NOT_FOUND	MediaTailor Impossible de transcoder l'annonce en raison d'un profil manquant à transcoder. En utilisant un ensemble d'annonces vide.
TRANSCODE_COMPLETED	Le transcodage vidéo est terminé. L'annonce peut être utilisée pour l'insertion d'annonces.
TRANSCODE_ERROR	MediaTailor a rencontré une erreur interne lors d'une tâche de transcodage. En utilisant un ensemble d'annonces vide.
TRANSCODE_IN_PROGRESS	Le transcodage vidéo est en cours. La vidéo transcodée n'est pas prête. En utilisant un ensemble d'annonces vide.

Propriétés des journaux de transcodage

Cette section décrit les propriétés des journaux de transcodage.

Propriété	Type	Obligatoire	Description
awsAccountId	chaîne	true	L'ID de AWS compte pour la MediaTailor configuration qui a été utilisée pour la session.
eventTimestamp	chaîne	true	Date et heure de l'événement.

Propriété	Type	Obligatoire	Description
originId	chaîne	true	Le nom de configuration indiqué dans la MediaTailor configuration. Cette source est différente de la source de contenu vidéo, qui fait également partie de la configuration.
eventType	chaîne	false	Code de l'événement qui a déclenché ce message de journal. Exemple: TRANSCODE_ERROR .
eventDescription	chaîne	false	Brève description de l'événement qui a déclenché ce message de journal, fournie par le MediaTailor service. Par défaut, celle-ci est vide.

Propriété	Type	Obligatoire	Description
<code>sessionId</code>	chaîne	false	L'identifiant numérique unique MediaTailor attribué à la session du joueur. Toutes les demandes effectuées par un joueur pour une session ont le même ID de session. Exemple: e039fd39-09f0-46b2-aca9-9871cc116cde .
<code>creativeUniqueId</code>	chaîne	false	L'identifiant unique de la création publicitaire qui est transcodée.
<code>profileName</code>	chaîne	false	
<code>adUri</code>	chaîne	false	L'URI de la création publicitaire.
<code>transcodeRequestId</code>	chaîne	false	L'identifiant unique de cette demande de transcodage.
<code>cacheStatus</code>	chaîne	false	Indique si l'annonce transcodée a été mise en MediaTailor cache.

Utiliser des journaux vendus pour envoyer AWS Elemental MediaTailor journaux

Vous pouvez utiliser des journaux distribués pour une plus grande flexibilité et pour contrôler l'endroit où vous souhaitez diffuser les journaux émis MediaTailor depuis votre configuration de lecture.

Avec les journaux distribués, MediaTailor envoie toutes les activités de journal associées à une configuration à Amazon CloudWatch Logs. CloudWatch Logs envoie ensuite le pourcentage de journaux que vous spécifiez vers la destination que vous avez choisie. Les destinations prises en charge sont un groupe de CloudWatch journaux Amazon Logs, un compartiment Amazon S3 ou un flux Amazon Data Firehose.

Étant donné que les journaux vendus sont disponibles à des prix réduits sur le volume, vous pourriez réaliser des économies par rapport à l'envoi direct de CloudWatch journaux à Logs. Pour en savoir plus sur les tarifs, consultez [Vended Logs](#) dans l'onglet [Logs](#) d'[Amazon CloudWatch Pricing](#).

Pour utiliser les journaux distribués, vous devez effectuer les opérations suivantes :

1. [Ajout d'autorisations](#).
2. [Création de destinations de livraison des journaux](#).
3. [Configurer la livraison des CloudWatch journaux dans Logs](#).
4. [Activer les connexions automatiques MediaTailor](#).

Pour plus d'informations sur les journaux distribués, consultez la section [Activer la journalisation à partir AWS des services](#) dans le guide de l'utilisateur CloudWatch des journaux. MediaTailor prend en charge la version 2 des journaux vendus.

Étape 1 : ajouter des autorisations pour la livraison des MediaTailor journaux

La personne qui configure les journaux vendus doit être autorisée à créer la destination de diffusion, à configurer la livraison des journaux et à activer les connexions distribuées. MediaTailor Utilisez les politiques suivantes pour vous assurer que vous disposez des autorisations appropriées pour configurer des journaux distribués.

Politiques relatives CloudWatch aux journaux et aux destinations de livraison

Les sections suivantes du guide de l'utilisateur d'Amazon CloudWatch Logs fournissent les politiques qui vous permettent de travailler avec les journaux enregistrés dans CloudWatch

les journaux et vos destinations de livraison. Si vous envoyez des journaux à plusieurs emplacements, vous pouvez combiner les déclarations de politique en une seule politique au lieu de créer plusieurs politiques.

- [Journaux envoyés à CloudWatch Logs](#)
- [Journaux envoyés à Amazon S3](#)
- [Journaux envoyés à Firehose](#)

Politique de configuration depuis la console

Si vous configurez la diffusion des journaux vendus via la console au lieu de l'API ou AWS CLI si vous devez disposer des autorisations supplémentaires suivantes dans votre politique.

Politique concernant les connexions distribuées MediaTailor

Pour créer, afficher ou modifier les journaux distribués dans MediaTailor, vous devez disposer des autorisations suivantes dans votre politique.

Pour plus d'informations sur l'ajout d'autorisations et l'utilisation des politiques, consultez [Gestion des identités et des accès pour AWS Elemental MediaTailor](#).

Étape 2 : Création de destinations de livraison pour les MediaTailor journaux

Créez les ressources auxquelles vos journaux seront envoyés. Enregistrez l'ARN de la ressource à utiliser lors de la configuration de la livraison du journal lors d'une étape ultérieure.

CloudWatch Destination de livraison du groupe de journaux

Utilisez l'une des méthodes suivantes pour vous aider à créer un groupe de journaux.

- Pour la console, voir [Créer un groupe de CloudWatch journaux](#) dans Logs du guide de l'utilisateur Amazon CloudWatch Logs.
- Pour l'API, consultez le [CreateLogGroup](#) manuel Amazon CloudWatch Logs API Reference.
- Pour les kits de développement logiciel et l'interface de ligne de commande, consultez la section [Utilisation CreateLogGroup avec un AWS SDK ou consultez le AWS CLI](#) guide de l'utilisateur Amazon CloudWatch Logs.

Destination de livraison du bucket Amazon S3

Utilisez l'une des méthodes suivantes pour vous aider à créer un compartiment.

- Pour la console, les kits de développement logiciel et l'interface de ligne de commande, voir [Créer un compartiment](#) dans le guide de l'utilisateur d'Amazon Simple Storage Service.

- Pour l'API, consultez [CreateBucket](#) le manuel Amazon Simple Storage Service API Reference.

Destination de livraison de Firehose Stream

Pour obtenir de l'aide sur la création d'un flux, consultez la section [Créer un flux Firehose depuis la console](#) dans le guide du développeur Amazon Data Firehose.

Étape 3 : activer les journaux distribués pour la configuration de MediaTailor lecture

Créez ou mettez à jour la configuration de lecture qui enverra les journaux à la destination de livraison que vous avez créée à l'étape précédente. Enregistrez le nom de la configuration à utiliser lors de la configuration de la livraison du journal lors d'une étape ultérieure.

- Pour activer les journaux distribués via la console, utilisez [Création d'une configuration](#) ou [Modification des paramètres de configuration](#) modifiez une configuration pour accéder aux paramètres de journalisation. Pour les stratégies de journalisation, choisissez Vended logs.
- Pour activer les journaux distribués via l'API, vous devez disposer d'une configuration existante. Utilisez `ConfigureLogsForPlaybackConfiguration` pour ajouter la stratégie de journalisation `Vended logs`.

Si vous utilisez l'ancienne stratégie de MediaTailor journalisation qui consiste à envoyer des journaux directement aux CloudWatch journaux et que vous souhaitez migrer vers des journaux vendus, consultez [Migration de la stratégie de journalisation](#).

Important

Si vous modifiez la stratégie de journalisation de Legacy CloudWatch à Vended Logs, cette modification MediaTailor sera effectuée dès que vous aurez enregistré les mises à jour. Vous ne recevrez plus de journaux tant que vous n'aurez pas complètement configuré la journalisation automatique.

Étape 4 : Configuration de la livraison des CloudWatch journaux dans Logs

Dans CloudWatch Logs, vous devez créer trois éléments pour représenter les éléments de la livraison des journaux. Ces éléments sont décrits en détail [CreateDelivery](#) dans le manuel Amazon

CloudWatch Logs API Reference. Les étapes de haut niveau pour configurer la diffusion à l'aide de l'API CloudWatch Logs sont les suivantes.

Pour configurer la livraison des CloudWatch journaux dans Logs (API)

1. Permet [PutDeliverySource](#) d'ajouter la source des journaux.

A `DeliverySource` représente la configuration de lecture qui génère les journaux. Vous avez besoin du nom de la configuration de lecture pour créer le `DeliverySource`.

2. Permet [PutDeliveryDestination](#) d'ajouter la destination où les journaux seront écrits.

A `DeliveryDestination` représente la destination de livraison. Vous avez besoin de l'ARN du groupe de journaux, du bucket ou du flux pour créer le `DeliveryDestination`.

3. À utiliser [PutDeliveryDestinationPolicy](#) si vous diffusez des journaux sur plusieurs comptes.

Si la destination de livraison se trouve dans un compte différent de celui de la configuration de lecture, vous avez besoin d'un `DeliveryDestinationPolicy`. Cette politique permet à CloudWatch Logs de fournir des journaux au `DeliveryDestination`.

4. Utilisez [CreateDelivery](#) pour lier le `DeliverySource` au `DeliveryDestination`.

A `Delivery` représente la connexion entre le `DeliverySource` et `DeliveryDestination`.

Migrer votre AWS Elemental MediaTailor stratégie de journalisation

Si vous modifiez la stratégie de journalisation de Legacy CloudWatch à Vended Logs, cette modification MediaTailor sera effectuée dès que vous aurez enregistré les mises à jour. Pour éviter toute interruption de votre flux de journalisation, procédez comme suit pour migrer votre stratégie de journalisation.

1. Suivez les étapes décrites dans la section [Utilisation de journaux vendus](#). Pour [Activer les connexions automatiques MediaTailor](#), sélectionnez les deux stratégies de journalisation (Vended logs et Legacy CloudWatch).

MediaTailor enverra les journaux à la fois via les journaux vendus et directement à CloudWatch Logs.

2. Apportez les modifications nécessaires à votre flux de travail en fonction de votre stratégie de journalisation et de votre destination de livraison.

3. Revisitez [Activer les connexions automatiques MediaTailor](#) et supprimez Legacy CloudWatch des stratégies de journalisation.

Écriture AWS Elemental MediaTailor se connecte directement à Amazon CloudWatch Logs

MediaTailor produit des journaux contenant des informations détaillées sur l'activité des sessions et les interactions avec le serveur de décision publicitaire, et les écrit sur Amazon CloudWatch. Les journaux fournissent une description séquentielle de l'activité qui se produit pendant la session.

MediaTailor peuvent également utiliser des grumes distribuées pour plus de flexibilité dans la livraison des grumes et des remises sur volume. Pour plus d'informations sur les journaux distribués, consultez [Utilisation de journaux vendus](#).

Rubriques

- [Autorisations pour Amazon CloudWatch Logs](#)
- [Journal « As Run » pour AWS Elemental MediaTailor Assemblage des canaux](#)
- [AWS Elemental MediaTailor Analyse des journaux ADS dans Amazon CloudWatch Logs Insights](#)

Autorisations pour Amazon CloudWatch Logs

Utilisez Gestion des identités et des accès AWS (IAM) pour créer un rôle donnant AWS Elemental MediaTailor accès à Amazon CloudWatch. Vous devez suivre ces étapes pour que CloudWatch les journaux soient publiés pour votre compte. CloudWatch publie automatiquement les statistiques de votre compte.

Pour autoriser MediaTailor l'accès à CloudWatch

1. Ouvrez la console IAM à l'adresse <https://console.aws.amazon.com/iam/>.
2. Dans le volet de navigation de la console IAM, sélectionnez Rôles, puis Créer un rôle.
3. Choisissez le type de rôle « Autre AWS compte ».
4. Dans le champ ID de compte, entrez votre identifiant de AWS compte.
5. Sélectionnez Demander un ID externe et entrez **Midas**. Cette option ajoute automatiquement une condition à la stratégie d'approbation qui permet au service d'assumer le rôle uniquement si la demande inclut l'élément `sts:ExternalId` approprié.

6. Choisissez Suivant : Autorisations.
7. Ajoutez une stratégie d'autorisations qui spécifie les actions que ce rôle peut effectuer. Sélectionnez l'une des options suivantes, puis choisissez Next : Review :
 - CloudWatchLogsFullAccess pour fournir un accès complet à Amazon CloudWatch Logs
 - CloudWatchFullAccess pour fournir un accès complet à Amazon CloudWatch
8. Pour Nom du rôle, saisissez **MediaTailorLogger**, puis choisissez Créer un rôle.
9. Sur la page Rôles, sélectionnez le rôle que vous venez de créer.
10. Modifiez la relation d'approbation pour mettre à jour le mandataire :
 1. Sur la page Récapitulatif du rôle, choisissez l'onglet Relation d'approbation.
 2. Choisissez Modifier la relation d'approbation.
 3. Dans le document de politique, remplacez le principal par le MediaTailor service. Elle doit ressembler à ce qui suit :

```
"Principal": {  
  "Service": "mediatailor.amazonaws.com"  
},
```

L'ensemble de la stratégie doit se présenter comme suit :

4. Choisissez Mettre à jour la politique d'approbation.

Journal « As Run » pour AWS Elemental MediaTailor Assemblage des canaux

Le journal As Run, dans le groupe de CloudWatch MediaTailor/Channel/AsRunLog journaux, affiche des informations sur les programmes et les pauses publicitaires au fur et à mesure de leur diffusion.

Lorsque vous créez une chaîne, le journal As Run est désactivé par défaut. À l'aide de la console ou du AWS CLI bouton AWS Command Line Interface (), vous pouvez activer et désactiver l'état du journal As Run pour chaque canal de votre compte.

Lorsque vous activez le journal As Run, crée MediaTailor automatiquement un rôle lié au service qui permet d'écrire et MediaTailor de gérer le journal As Run dans votre compte CloudWatch Logs. Pour plus d'informations sur les rôles liés à un service, consultez [Utilisation de rôles liés à des services pour MediaTailor](#).

 Note

Le journal As Run ne prend actuellement en charge que le programme par défaut. Pour l'instant, il ne prend pas en charge l'AlternateMedia créé par les règles du programme. Cela signifie qu'il ne génère actuellement pas le journal As Run pour AlternateMedia.

Rubriques

- [Activation du journal As Run](#)
- [Désactivation du journal As Run](#)

Activation du journal As Run

Pour activer le journal As Run, spécifiez le nom du canal et activez le type de journal As Run pour ce canal.

Console

Pour activer le journal As Run lors de la création d'une chaîne

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez Assemblage de canaux > Canaux.
3. Dans la barre de navigation, choisissez Créer une chaîne.
4. Dans les volets Définir les détails du canal, Configurer les sorties et Contrôle d'accès, configurez votre canal comme vous le souhaitez.
5. Dans le volet de contrôle d'accès, choisissez Next.
6. Dans le volet Journalisation, sous Types de journaux, sélectionnez Activer en tant qu'exécution pour activer le journal en cours d'exécution.

Pour activer le journal As Run lors de la mise à jour d'une chaîne

Note

Si la chaîne est en cours de diffusion, vous devez d'abord arrêter cette chaîne avant de pouvoir la mettre à jour. Après avoir arrêté la chaîne, vous pouvez choisir **Actions > Modifier** pour commencer à mettre à jour la chaîne.

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux > Canaux**.
3. Choisissez la chaîne que vous souhaitez mettre à jour pour activer le journal As Run.
4. Sélectionnez **Actions > Edit (Modifier)**.
5. Dans les volets **Définir les détails du canal**, **Configurer les sorties** et **Contrôle d'accès**, mettez à jour la configuration de votre canal comme vous le souhaitez.
6. Dans le volet de contrôle d'accès, choisissez **Next**.
7. Dans le volet **Journalisation**, sous **Types de journaux**, sélectionnez **Activer en tant qu'exécution** pour activer le journal en cours d'exécution.

Pour activer le journal As Run à partir du Logging onglet

Note

Si la chaîne est en cours d'exécution, vous devez utiliser l'onglet **Journalisation** au lieu de choisir **Actions > Modifier** pour activer le journal As Run.

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux > Canaux**.
3. Choisissez la chaîne pour laquelle vous souhaitez activer le journal As Run.
4. Dans la barre de navigation située sous le nom de la chaîne, choisissez **Logging**.
5. Sous **Journalisation > Types de journaux**, sélectionnez **En cours d'exécution** pour activer le journal En cours d'exécution.

AWS Command Line Interface (AWS CLI)

Pour activer le journal As Run

Exécutez la [commande](#) `configure-logs-for-channel` et spécifiez les valeurs appropriées pour les paramètres requis.

Cet exemple est formaté pour Linux, macOS ou Unix et utilise le caractère de continuation de ligne barre oblique inverse (`\`) pour améliorer la lisibilité.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types AS_RUN
```

Cet exemple est mis en forme pour Microsoft Windows et utilise le caractère de continuation de ligne caret (`^`) pour améliorer la lisibilité.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types AS_RUN
```

Où :

- *MyChannel* est le nom de la chaîne dont vous êtes propriétaire et pour laquelle vous souhaitez activer le journal As Run.

Si la commande s'exécute correctement, vous recevez une sortie similaire à ce qui suit.

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": [  
    "AS_RUN"  
  ]  
}
```

Désactivation du journal As Run

Pour désactiver le journal As Run pour un canal sur lequel il est activé, spécifiez le nom du canal et désactivez le type de journal As Run pour ce canal.

Console

Pour désactiver le journal As Run lors de la mise à jour d'une chaîne

Note

Si la chaîne est en cours de diffusion, vous devez d'abord arrêter cette chaîne avant de pouvoir la mettre à jour. Après avoir arrêté la chaîne, vous pouvez choisir **Actions > Modifier** pour commencer à mettre à jour la chaîne.

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux > Canaux**.
3. Choisissez la chaîne que vous souhaitez mettre à jour pour activer le journal As Run.
4. Sélectionnez **Actions > Edit (Modifier)**.
5. Dans les volets **Définir les détails du canal**, **Configurer les sorties** et **Contrôle d'accès**, mettez à jour la configuration de votre canal comme vous le souhaitez.
6. Dans le volet de contrôle d'accès, choisissez **Next**.
7. Dans le volet **Journalisation**, sous **Types de journaux**, désactivez **Activer en cours d'exécution** pour désactiver le journal en cours d'exécution.

Pour désactiver le journal As Run à partir du Logging onglet

Note

Si la chaîne est en cours d'exécution, vous devez utiliser l'onglet **Journalisation** au lieu de choisir **Actions > Modifier** pour désactiver le journal As Run.

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet de navigation, choisissez **Assemblage de canaux > Canaux**.
3. Choisissez la chaîne pour laquelle vous souhaitez désactiver le journal As Run.
4. Dans la barre de navigation sous le nom de la chaîne, choisissez **Logging**.

5. Sous **Journalisation > Types de journaux**, désactivez **En cours d'exécution** pour désactiver le journal **En cours d'exécution**.

AWS Command Line Interface (AWS CLI)

Pour désactiver le journal **As Run**

Exécutez la [commande](#) `configure-logs-for-channel` et spécifiez les valeurs appropriées pour les paramètres requis.

Cet exemple est formaté pour Linux, macOS ou Unix et utilise le caractère de continuation de ligne barre oblique inverse (\) pour améliorer la lisibilité.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types
```

Cet exemple est mis en forme pour Microsoft Windows et utilise le caractère de continuation de ligne caret (^) pour améliorer la lisibilité.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types
```

Où :

- *MyChannel* est le nom de la chaîne dont vous êtes propriétaire et pour laquelle vous souhaitez désactiver le journal **As Run**.

Si la commande s'exécute correctement, vous recevez une sortie similaire à ce qui suit.

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": []  
}
```

AWS Elemental MediaTailor Analyse des journaux ADS dans Amazon CloudWatch Logs Insights

Vous pouvez consulter et interroger les journaux du serveur de décision AWS Elemental MediaTailor publicitaire (ADS) à l'aide d'Amazon CloudWatch Logs Insights. MediaTailor envoie des journaux d'événements à CloudWatch pour le traitement normal et les conditions d'erreur. Les journaux respectent un schéma JSON. Grâce à CloudWatch Logs Insights, vous pouvez sélectionner des journaux par période, puis exécuter des requêtes sur ceux-ci.

Pour des informations générales, voir [Analyser les données des CloudWatch journaux avec Logs Insights](#).

Note

Pour accéder aux journaux, vous devez disposer des autorisations nécessaires pour accéder à Amazon CloudWatch. Pour obtenir des instructions, veuillez consulter [Autorisations pour Amazon CloudWatch Logs](#).

Pour consulter et interroger les journaux ADS à l'aide de la CloudWatch console

1. Ouvrez la CloudWatch console à l'adresse <https://console.aws.amazon.com/cloudwatch/>.
2. Dans le volet de navigation, sous Logs (Journaux), choisissez Insights.
3. Dans la barre de recherche **AdDec**, saisissez, puis dans la liste déroulante, sélectionnez `MediaTailor/AdDecisionServerInteractions`.
4. (Facultatif) Ajustez la période que vous souhaitez étudier.
5. (Facultatif) Modifiez la requête dans la boîte de dialogue. Pour obtenir des conseils généraux, consultez la section Syntaxe des requêtes [CloudWatch Logs Insights](#). Pour obtenir des exemples de requêtes pour ADS MediaTailor, consultez [Interrogation des journaux ADS](#).
6. Choisissez Exécuter la requête. La requête peut prendre quelques secondes, pendant lesquelles Cancel (Annuler) apparaît à la place de Run query (Exécuter la requête).
7. (Facultatif) Pour exporter les résultats sous forme de fichier CSV, choisissez Actions, puis Download query results (CSV) (Télécharger les résultats de la requête - CSV).

 Note

La console limite le nombre d'enregistrements qu'elle renvoie dans les résultats des requêtes et qu'elle exporte. Par conséquent, pour les données en masse, utilisez l'API, le AWS Command Line Interface (AWS CLI) ou un SDK.

Rubriques

- [Interrogation des journaux ADS](#)

Interrogation des journaux ADS

CloudWatch Logs Insights propose un ensemble complet d'options pour interroger vos journaux. Pour des informations détaillées sur la syntaxe des requêtes, consultez la section Syntaxe des requêtes de [CloudWatch Logs Insights](#). Cette section fournit des exemples de requêtes courantes pour vous aider à démarrer avec vos requêtes de journaux ADS. Toutes les requêtes s'exécutent sur les journaux pour le paramètre de plage de temps actuel.

La requête suivante récupère toutes les informations des journaux ADS.

```
fields @timestamp, eventType, sessionId, requestId, @message
| sort sessionId, @timestamp asc
```

La requête suivante récupère toutes les demandes adressées à ADS. Cette requête indique un moyen de récupérer le contenu de l'en-tête de demande pour les MediaTailor journaux.

```
fields @timestamp, adsRequestUrl, requestHeaders.0.value as @userAgent,
requestHeaders.1.value as @xForwardedFor, sessionId, requestId
| filter eventType = "MAKING_ADS_REQUEST"
| sort @timestamp asc
```

La requête suivante permet de récupérer les annonces MediaTailor insérées pour une session donnée.

```
fields @timestamp, sessionId, requestId, @message
| filter eventType = "FILLED_AVAIL"
| sort @timestamp asc
```


La requête suivante permet de récupérer les URL de suivi que MediaTailor ont appelé au nom du joueur.

```
fields @timestamp, beaconInfo.trackingEvent, beaconInfo.beaconUri,  
  beaconInfo.headers.0.value as @userAgent, beaconInfo.headers.1.value as  
  @xForwardedFor, sessionId, requestId  
| filter eventType = "BEACON_FIRED"  
| sort @timestamp asc
```

La requête suivante récupère des informations pour une session de lecture spécifique, en filtrant les résultats par sessionId.

```
fields @timestamp, eventType, sessionId, requestId, @message  
| filter sessionId = "0aaf6507-c6f9-4884-bfe7-f2f841cb8195"  
| sort @timestamp asc
```

La requête suivante récupère des informations pour une seule demande, en filtrant les résultats par requestId.

```
fields @timestamp, eventType, sessionId, requestId, @message  
| filter requestId = "f5d3cf39-6258-4cf1-b3f6-a34ff8bf641d"  
| sort @timestamp asc
```

La requête suivante récupère un nombre d'entrées de journal pour chaque type d'événement qui a été consigné.

```
fields eventType  
| stats count() as @eventCount by eventType
```

La requête suivante récupère l'ID de diffusion et la liste des publicités ignorées pour toutes les diffusions avec des publicités ignorées.

```
fields avail.availId  
| parse @message '"skippedAds":[*]' as @skippedAdsList  
| filter ispresent(@skippedAdsList)
```

Contrôler le volume de AWS Elemental MediaTailor journaux

MediaTailor les journaux des sessions d'insertion d'annonces sont parfois détaillés. Pour réduire les coûts liés aux journaux, vous pouvez définir le pourcentage de journaux de session MediaTailor

envoyés à Amazon CloudWatch Logs. Par exemple, si votre configuration de lecture comporte 1 000 sessions d'insertion d'annonces et que vous définissez un pourcentage activé de 60, MediaTailor envoie les journaux de 600 sessions à CloudWatch Logs. MediaTailor décide au hasard pour laquelle des sessions envoyer des journaux. Si vous souhaitez consulter les journaux d'une session spécifique, vous pouvez utiliser le mode [journal de débogage](#).

Lorsque vous définissez un pourcentage de journalisation, crée MediaTailor automatiquement un rôle lié à un service qui accorde MediaTailor les autorisations nécessaires pour écrire CloudWatch des journaux sur votre compte. Pour plus d'informations sur l'utilisation des rôles liés aux services, consultez [Utilisation de rôles liés à des services pour MediaTailor](#)

Création d'une configuration de journal

Pour contrôler le pourcentage de journaux de session que MediaTailor écrivent dans les CloudWatch journaux, vous devez créer une configuration de journal pour votre configuration de lecture. Lorsque vous créez une configuration de journal, vous spécifiez un nom de configuration de lecture et une valeur de pourcentage activée.

Console

Pour créer une configuration de journal pour une configuration de lecture existante

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet Configuration de la lecture, sélectionnez la configuration de lecture pour laquelle vous souhaitez définir la configuration du journal.
3. Choisissez Modifier.
4. Dans Configuration du journal, spécifiez une valeur de pourcentage activée.

Pour créer une configuration de journal pour une nouvelle configuration de lecture

- Suivez la procédure décrite dans [Configuration du journal](#).

AWS Command Line Interface (AWS CLI)

Pour créer une configuration de journal pour une configuration de lecture existante

Pour créer une configuration de journal à l'aide de AWS CLI, exécutez la [commande](#) `configure-logs-for-playback-configuration` et spécifiez les valeurs appropriées pour les paramètres requis.

Cet exemple est formaté pour Linux, macOS ou Unix et utilise le caractère de continuation de ligne barre oblique inverse (\) pour améliorer la lisibilité.

```
$ aws mediatailor configure-logs-for-playback-configuration \  
--percent-enabled 10 \  
--playback-configuration-name MyPlaybackConfiguration
```

Cet exemple est mis en forme pour Microsoft Windows et utilise le caractère de continuation de ligne caret (^) pour améliorer la lisibilité.

```
C:\> aws mediatailor configure-logs-for-playback-configuration ^  
--percent-enabled 10 ^  
--playback-configuration-name MyPlaybackConfiguration
```

Où :

- *percent-enabled* est le pourcentage des journaux de session de configuration de lecture MediaTailor envoyés à CloudWatch Logs.
- *playback-configuration-name* est le nom de la configuration de lecture pour laquelle définir les paramètres de configuration du journal.

Si la commande s'exécute correctement, vous recevez une sortie similaire à ce qui suit.

```
{  
  "PercentEnabled": 10,  
  "PlaybackConfigurationName": "MyPlaybackConfiguration"  
}
```

Pour créer une configuration de journal pour un nouveau configuration de lecture

- Utilisez l'`configure-logs-for-playback-configuration` option de la commande [put-playback-configuration](#).

Désactivation de la configuration d'un journal

Une fois que vous avez créé une configuration de journal, vous ne pouvez pas la supprimer. Vous pouvez uniquement la désactiver. Pour désactiver la configuration du journal, définissez la

valeur du pourcentage activé sur 0 à l'aide de la MediaTailor console ou de l'API. Cela désactive l'enregistrement de toutes les sessions pour cette configuration de lecture.

Si vous souhaitez supprimer le rôle lié à un service MediaTailor utilisé pour la ou les configurations de journal de votre compte, vous devez d'abord désactiver toutes vos configurations de journal. Pour plus d'informations sur la façon de supprimer le rôle lié à un service, consultez [Utilisation de rôles liés à des services pour MediaTailor](#)

Console

Pour désactiver la configuration du journal sur une configuration de lecture

1. Connectez-vous à Console de gestion AWS et ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Dans le volet Configuration de la lecture, sélectionnez la configuration de lecture pour laquelle vous souhaitez désactiver la configuration du journal.
3. Choisissez Modifier.
4. Dans Configuration du journal, définissez la valeur du pourcentage activé sur 0. Cela désactive l'enregistrement de toutes les sessions pour cette configuration de lecture.
5. Cliquez sur Enregistrer.

AWS Command Line Interface (AWS CLI)

Pour désactiver la configuration d'un journal

- Définissez la percent-enabled valeur à l'0 aide de la commande <https://docs.aws.amazon.com/cli/latest/reference/mediatailor/configure-logs-for-playback-configuration.html> configure-logs-for-playback-configuration.

Filtrage AWS Elemental MediaTailor journaux et événements

Les journaux émis par une configuration de lecture MediaTailor incluent des informations sur diverses activités qui se produisent pendant la session de lecture. Ces activités sont identifiées dans le type d'événement des journaux. De nombreux événements sont enregistrés par défaut. Pour vous aider à contrôler le coût des journaux sur Amazon CloudWatch, vous pouvez spécifier les journaux qui sont MediaTailor émis.

MediaTailor vous permet de contrôler le filtrage des journaux afin que vous puissiez effectuer les opérations suivantes :

- Spécifiez les événements de journal que vous souhaitez exclure des journaux
- Activer la journalisation des réponses brutes depuis le serveur de décision publicitaire (ADS)

Vous pouvez définir ces préférences de filtrage des journaux indépendamment pour chaque session de lecture, ou par défaut pour toutes les sessions de lecture pour une configuration de lecture.

- Pour filtrer les journaux par session, incluez les paramètres de requête dans la demande d'initialisation de la session de lecture.
- Pour filtrer les journaux en fonction de la configuration de lecture, utilisez la MediaTailor console ou l'API pour indiquer vos préférences dans les paramètres de configuration de lecture.

Les sections suivantes fournissent des instructions pour activer le filtrage des journaux sur les sessions et les configurations de lecture.

Per-session filtres de journal

Pour définir un niveau personnalisé de détails du journal pour chaque session, ajoutez les paramètres suivants à votre demande initiale de session de lecture côté serveur ou côté client. Ajoutez des valeurs aux paramètres pour représenter les événements que vous souhaitez inclure ou exclure, dans un format séparé par des virgules :

- `aws.adsInteractionLogPublishOptInEventTypes` pour recevoir les journaux relatifs à des interactions spécifiques avec le serveur de décision publicitaire (ADS).
- `aws.adsInteractionLogExcludeEventTypes` pour ne plus recevoir les journaux relatifs à des interactions ADS spécifiques.
- `aws.manifestServiceLogExcludeEventTypes` pour arrêter de recevoir les journaux relatifs à des interactions spécifiques avec les services de manifeste.

Pour obtenir la liste des types de journaux et d'événements que MediaTailor émettent, reportez-vous aux [Journaux des manifestes](#) sections [Journaux ADS](#), et [Journaux de transcodage](#).

Si vous ne transmettez aucun paramètre de requête pour le filtrage des journaux, MediaTailor écrit tous les journaux vers votre destination de livraison.

Exemple initialisation de session côté serveur à l'aide de filtres de journalisation

Pour exclure GENERATED_MANIFEST_PARSING_ERROR des événements de vos journaux manifestes et MAKING_ADS_REQUEST des journaux ADS, la demande d'initialisation de session devrait se présenter comme suit :

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?  
aws.logMode=DEBUG&aws.manifestServiceLogExcludeEventTypes=GENERATED_MANIFEST,PARSING_ERROR&aws.
```

Pour activer les journaux bruts à partir de votre ADS, incluez la RAW_ADS_RESPONSE valeur du AdsInteractionPublishOptInEventType paramètre :

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?  
aws.adsInteractionPublishOptInEventType=RAW_ADS_RESPONSE
```

Exemple initialisation de session côté client avec filtres de journalisation

Pour exclure les événements de journal lors de l'initialisation de la session côté client, incluez availSuppression et enregistrez les paramètres dans la requête POST de votre client adressée à MediaTailor. Pour plus d'informations sur la création d'une demande de session de lecture côté client, consultez [Client-side suivi des publicités](#). L'exemple suivant exclut CONFIG_SECURITY_ERROR des PARSING_ERROR événements de vos journaux manifestes et MAKING_ADS_REQUEST des journaux ADS.

```
POST parent.m3u8  
{  
  "adsInteractionLog": {  
    ...  
    "excludeEventTypes": [  
      "MAKING_ADS_REQUEST"  
    ]  
  },  
  "manifestServiceLog": {  
    ...  
    "excludeEventTypes": [  
      "GENERATED_MANIFEST",  
      "PARSING_ERROR"  
    ]  
  },  
  "logMode": "DEBUG"
```

```
}
```

Pour activer les journaux bruts à partir de votre ADS, incluez la `RAW_ADS_RESPONSE` valeur du `publishOptInEventTypes` paramètre :

```
POST parent.m3u8
{
  "adsInteractionLog": {
    "publishOptInEventTypes": ["RAW_ADS_RESPONSE"],
    "excludeEventTypes": [
      "MAKING_ADS_REQUEST"
    ]
  },
  "manifestServiceLog": {
    ...
    "excludeEventTypes": [
      "GENERATED_MANIFEST",
      "PARSING_ERROR"
    ]
  },
  "logMode": "DEBUG"
}
```

Per-playback filtres du journal de configuration

Utilisez les paramètres de la configuration de lecture pour définir les types d'événements de journal qui MediaTailor sont émis par défaut à partir de cette configuration de lecture. MediaTailor utilise ces paramètres de filtrage des journaux par défaut pour toutes les sessions qui n'incluent pas de paramètres de requête de filtrage dans la demande d'initialisation de session.

Vous pouvez choisir de procéder comme suit :

- Recevez des journaux pour des interactions spécifiques avec le serveur de décision publicitaire (ADS).
- Excluez les journaux pour des interactions ADS spécifiques.
- Excluez les journaux pour des interactions spécifiques avec les services de manifeste.

Pour définir ces paramètres depuis la MediaTailor console, consultez [Création d'une configuration](#).

Pour l' MediaTailor API, voir [PutPlaybackConfiguration](#) la référence de l'AWS Elemental MediaTailor API.

Pour obtenir la liste des types de journaux et d'événements que MediaTailor émettent, reportez-vous aux [Journaux des manifestes](#) sections [Journaux ADS](#), et [Journaux de transcodage](#).

Génération AWS Elemental MediaTailor journaux de débogage

Utilisez les journaux de débogage pour résoudre les problèmes de session de lecture liés à l'insertion d' MediaTailor annonces. Pour générer des journaux de débogage, configurez le mode de journalisation sur débogage à la demande du joueur. MediaTailor Pour les rapports côté serveur, définissez le mode journal dans la demande de lecture. Pour les rapports côté client, définissez le mode de journalisation dans la demande d'initialisation de session.

Lorsque le mode journal est défini sur débogage, MediaTailor écrit tous les types d'événements de journal dans CloudWatch Logs. Les journaux fournissent des informations sur les événements suivants. Pour une liste complète des données produites dans les journaux de débogage, voir Champs du journal de [débogage](#).

- Interaction d'origine : détails sur MediaTailor les interactions avec le serveur d'origine. Par exemple, la réponse du manifeste d'origine, le type de manifeste et l'URL d'origine.
- Manifeste généré : détails sur la réponse à la session de lecture provenant de MediaTailor. Par exemple, le manifeste qui est MediaTailor généré.
- Session initialisée : détails relatifs à l'initialisation de la session, tels que l'ID de session.

Pour personnaliser les types d'événements de journal que vous recevez par session, consultez [Filtrage des journaux et des événements](#).

Conditions préalables

Pour configurer le mode journal sur débogage, vous devez d'abord MediaTailor autoriser l'envoi de journaux à CloudWatch, si ce n'est pas déjà fait. Une fois que vous avez accordé l'autorisation MediaTailor d'accès CloudWatch, vous êtes prêt à activer le mode journal de débogage. Pour plus d'informations sur la façon d'accorder MediaTailor l'autorisation d'accès, CloudWatch consultez [Configuration des autorisations pour Amazon CloudWatch](#).

Comment configurer le mode journal pour le débogage

Cette section explique comment configurer le mode de journalisation pour le débogage pour les rapports côté serveur et les rapports côté client.

Server-side rapports

Pour les rapports côté serveur, incluez le paramètre et la valeur de la `?aws.logMode=DEBUG` requête dans la demande de GET HTTP lecture de votre lecteur au point de terminaison HLS ou DASH. MediaTailor Pour des informations générales sur les rapports côté serveur, consultez la section [Server-side Rapports](#).

Important

La valeur DEBUG est sensible à la casse.

Une demande de lecture qui inclut `?aws.logMode=DEBUG` se présente comme suit :

Exemple Demande de lecture vers un point de terminaison HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.logMode=DEBUG
```

Après avoir défini le mode de journalisation sur débogage, nous vous recommandons de vérifier que la session de journalisation du débogage est active. Pour vérifier que la session de débogage est active, vérifiez s'il existe des CloudWatch journaux pour l'ID de session. L'ID de session est inclus dans le point de terminaison de lecture MediaTailor fourni. Pour de plus amples informations, veuillez consulter [Verify that the debug log mode is active for your playback session](#).

Client-side rapports

Pour les rapports côté client, incluez la `logMode` clé et la `DEBUG` valeur dans le corps de la demande d'initialisation de POST HTTP session de votre client au point de terminaison `/v1/session`. MediaTailor Pour des informations générales sur les rapports côté client, voir [Client-Side Rapports](#).

Important

La valeur DEBUG est sensible à la casse.

Après avoir défini le mode de journalisation sur débogage, nous vous recommandons de vérifier que la session de débogage est active. Pour vérifier que la session de débogage est active, vérifiez qu'un

SESSION_INITIALIZED événement est associé à l'ID de session dans les CloudWatch journaux. L'ID de session est inclus dans le point de terminaison de lecture MediaTailor fourni. Pour de plus amples informations, veuillez consulter [Verify that the debug log mode is active for your playback session](#).

Nombre maximal de sessions de débogage actives

Vous pouvez avoir un maximum de 10 sessions de journal de débogage actives. Lorsque votre lecteur envoie sa demande d'initialisation de session ou de lecture à MediaTailor, MediaTailor vérifie si la limite a été atteinte. Si c'est le cas, MediaTailor vérifie s'il existe des sessions périmées. Une session est obsolète si elle n'a pas été consultée pendant un certain temps. Pour les diffusions en direct, cette durée est de 10 minutes, pour les diffusions en VOD, elle est de 30 minutes.

Si la limite maximale de sessions de journal de débogage actives a été atteinte, les journaux de débogage ne sont pas écrits dans CloudWatch les journaux de votre session. Si vous ne voyez pas les journaux de débogage dans CloudWatch les journaux de votre session, vous avez peut-être atteint cette limite. Pour vérifier si la limite a été atteinte, consultez [Verify that the debug log mode is active for your playback session](#).

Champs du journal de débogage

Le tableau suivant répertorie les champs du journal de débogage dans lesquels vous MediaTailor écrivez. CloudWatch

Champ	Description
awsAccountId	Votre Compte AWS carte d'identité.
customerId	Votre numéro de MediaTailor client.
eventTimestamp	Horodatage ISO 8601 associé à l'événement du journal de débogage.
eventType	Type d'événement du journal de débogage. Valeurs : • ORIGIN_INTERACTION — Détails sur MediaTailor les interactions avec le serveur d'origine. Par exemple, la réponse du manifeste d'origine, le type de manifeste et l'URL d'origine.

Champ	Description
	• GENERATED_MANIFEST — Informations sur la réponse à la session de lecture provenant de MediaTailor. Par exemple, le manifeste qui est MediaTailor généré. • SESSION_INITIALIZED — Détails d'initialisation de session, tels que l'ID de session.
<code>originRequestUrl</code>	URL de votre serveur d'origine qui est récupérée pour cette demande.
<code>mediaTailorPath</code>	Le MediaTailor point de terminaison qui a été appelé, y compris tous les paramètres transmis MediaTailor dans la demande de manifeste initiale.
<code>requestId</code>	L'ID d'une requête HTTP spécifique adressée à MediaTailor.
<code>responseBody</code>	Le manifeste dans le corps de la réponse provenant de MediaTailor. Il s'agit soit du manifeste d'origine brut, soit du manifeste généré par MediaTailor.
<code>sessionId</code>	L'ID de session de lecture.
<code>sessionType</code>	Type de session de lecture. Valeurs: HLS, DASH

Lire les journaux de débogage

MediaTailor écrit les journaux de débogage dans Amazon CloudWatch Logs. Des frais de CloudWatch journalisation typiques s'appliquent. Utilisez CloudWatch Insights pour lire les journaux de débogage. Pour plus d'informations sur l'utilisation de CloudWatch Logs Insights, consultez la section [Analyse des données des CloudWatch journaux avec Logs Insights](#) dans le guide de l'utilisateur d'AWS CloudWatch Logs.

Note

L'affichage des journaux de débogage peut prendre quelques minutes. CloudWatch Si vous ne voyez pas les journaux, patientez quelques minutes et réessayez. Si vous ne voyez

toujours pas les journaux, il se peut que vous ayez atteint le nombre maximum de sessions de journal de débogage actives. Pour vérifier si c'est le cas, exécutez une CloudWatch requête pour voir si une session de débogage a été initialisée pour votre session de lecture. Pour de plus amples informations, veuillez consulter [Verify that the debug log mode is active for your playback session](#).

Exemples

Cette section inclut des exemples de requêtes que vous pouvez utiliser pour lire les données du journal de MediaTailor débogage.

Exemple 1 : Vérifiez que le mode journal de débogage est actif pour votre session de lecture

```
fields @timestamp, @message
| filter sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
| filter eventType = "SESSION_INITIALIZED" # client-side reporting
or mediaTailorPath like "/v1/master" # server-side reporting HLS
or mediaTailorPath like "/v1/dash" # server-side reporting DASH
```

Exemple 2 : Afficher les réponses depuis votre origine

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter eventType = "ORIGIN_MANIFEST" and sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Exemple 3 : Afficher le manifeste généré par MediaTailor pour une session donnée

```
fields @timestamp, responseBody, @message
| filter mediaTailorPath like "/v1/master/" and eventType = "GENERATED_MANIFEST" and
sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Exemple 4 : Afficher tous les événements pour un ID de **demande** **donné**

Utilisez cette requête pour afficher le manifeste d'origine et le manifeste généré par MediaTailor.

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter requestId = "e5ba82a5-f8ac-4efb-88a0-55bed21c45b4"
```

Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics

Vous pouvez surveiller les AWS Elemental MediaTailor métriques à l'aide de CloudWatch.

CloudWatch collecte des données brutes sur les performances du service et traite ces données sous forme de mesures 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 plus d'informations, consultez le guide de CloudWatch l'utilisateur [Amazon](#).

Les métriques peuvent être utiles lorsque vous examinez des manifestes périmés. Pour de plus amples informations, veuillez consulter [Utilisation de métriques pour diagnostiquer les manifestes périmés](#).

Les métriques sont d'abord regroupées par espace de noms de service, puis par les différentes combinaisons de dimension au sein de chaque espace de noms.

Pour afficher les statistiques à l'aide de la CloudWatch console

1. Ouvrez la CloudWatch console à l'adresse <https://console.aws.amazon.com/cloudwatch/>.
2. Dans le panneau de navigation, sélectionnez Métriques.
3. Sous Toutes les métriques, choisissez l'espace de MediaTailor noms.
4. Sélectionnez la dimension de métrique pour afficher les métriques (par exemple, pour originID).
5. Spécifiez la période que vous souhaitez afficher.

Pour afficher les statistiques à l'aide du AWS Command Line Interface (AWS CLI)

- À partir d'une invite de commande, utilisez la commande suivante :

```
aws cloudwatch list-metrics --namespace "AWS/MediaTailor"
```

AWS Elemental MediaTailor CloudWatch métriques

L'espace de AWS Elemental MediaTailor noms inclut les métriques suivantes. Ces métriques sont publiées par défaut sur votre compte.

Métriques de Channel Assembly (CA)

Dans le tableau suivant, toutes les mesures sont disponibles par canal ou par sortie de canal.

Métrique	Description
<code>4xxErrorCount</code>	Le nombre d'4xxerreurs.
<code>5xxErrorCount</code>	Le nombre d'5xxerreurs.
<code>RequestCount</code>	Nombre total de demandes . Le nombre de transactions dépend en grande partie de la fréquence à laquelle les joueurs demandent des manifestes mis à jour et du nombre de joueurs. Chaque demande de lecteur est comptabilisée comme une transaction.
<code>TotalTime</code>	Le temps nécessaire au serveur d'applications pour traiter la demande, y compris le temps nécessaire pour recevoir des octets en provenance du client et du réseau et pour écrire des octets dans ces derniers.

Server-side Ad-insertion Métriques (SSAI)

Le tableau suivant répertorie les mesures d'insertion de publicités côté serveur.

Métrique	Description
<code>AdDecisionServer.Ads</code>	Nombre d'annonces incluses dans les réponses du serveur de décision publicitaire (ADS) au cours de la CloudWatch période que vous avez spécifiée.

Métrique	Description
	Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>AdDecisionServer.Duration</code>	Durée totale, en millisecondes, de toutes les publicités MediaTailor reçues de l'ADS au cours de la CloudWatch période que vous avez spécifiée. Cette durée peut être supérieure à celle <code>Avail.Duration</code> que vous avez spécifiée. Dimensions : <code>ConfigurationName</code>
<code>AdDecisionServer.Errors</code>	Le nombre de réponses par code d'état non HTTP 200, de réponses vides et de réponses expirées MediaTailor reçues de l'ADS au cours de la CloudWatch période que vous avez spécifiée. Dimensions : <code>ConfigurationName</code>
<code>AdDecisionServer.FillRate</code>	Moyenne simple de la vitesse à laquelle les réponses de l'ADS ont rempli les diffusions de publicités individuelles correspondantes pour la période que vous avez spécifiée. Pour obtenir la moyenne pondérée, calculez <code>AdDecisionServer.Duration</code> sous forme de pourcentage de <code>Avail.Duration</code> . Pour plus d'informations sur les moyennes simples et pondérées, consultez Moyennes simples et pondérées. Dimensions : <code>ConfigurationName</code>

Métrique	Description
<code>AdDecisionServer.Latency</code>	Temps de réponse en millisecondes pour les demandes adressées MediaTailor à l'ADS. Dimensions : <code>ConfigurationName</code>
<code>AdDecisionServer.Timeouts</code>	Le nombre de demandes adressées à l'ADS avec un délai d'expiration au cours de la CloudWatch période que vous avez spécifiée. Dimensions : <code>ConfigurationName</code>
<code>AdNotReady</code>	Nombre de fois où l'ADS a pointé vers une publicité non encore transcodée par le service de transcodeur interne dans la période que vous avez spécifiée. Une valeur élevée pour cette métrique peut contribuer à un <code>Avail.FillRate</code> d'ensemble faible. Dimensions : <code>ConfigurationName</code>
<code>AdsBilled</code>	Le nombre d'annonces pour lesquelles les clients MediaTailor sont facturés en fonction de leur insertion. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>Avail.Duration</code>	Le nombre total prévu de millisecondes d'annonce disponible au cours de cette période. CloudWatch Le total prévu est basé sur les durées de disponibilité indiquées dans le manifeste d'origine. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>Avail.FilledDuration</code>	Le nombre prévu de millisecondes de temps de disponibilité des publicités qui MediaTailor seront remplies d'annonces au cours de cette période. CloudWatch Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>Avail.FillRate</code>	La moyenne simple prévue des taux auxquels les annonces individuelles MediaTailor seront traitées sera disponible au cours de la CloudWatch période. Pour obtenir la moyenne pondérée, calculez <code>Avail.FilledDuration</code> sous forme de pourcentage de <code>Avail.Duration</code> . Pour plus d'informations sur les moyennes simples et pondérées, consultez Moyennes simples et pondérées. Le <code>Avail.FillRate</code> maximal que MediaTailor peut atteindre est délimité par <code>AdDecisionServer.FillRate</code> . Si <code>Avail.FillRate</code> est faible, comparez-le à <code>AdDecisionServer.FillRate</code> . Si <code>AdDecisionServer.FillRate</code> est faible, votre ADS peut ne pas renvoyer suffisamment de publicités pour les durées de diffusion. Dimensions : <code>ConfigurationName</code>

Métrique	Description
<code>Avail.Impression</code>	Nombre de publicités pour lesquelles des balises d'impression MediaTailor ont été déclenchées lors de la création de rapports côté serveur. MediaTailor émet un compte pour chaque annonce comportant au moins un URI d'impression, que ces balises aient été diffusées ou non. Utilisez cette métrique comme dénominateur pour calculer le taux d'achèvement des vidéos (magnétoscope) (voir <code>Avail.Complete</code>). Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>Avail.Complete</code>	Nombre de publicités pour lesquelles des balises complètes MediaTailor ont été déclenchées lors de la génération de rapports côté serveur. MediaTailor émet un compte pour chaque annonce qui possède au moins un URI de suivi complet, que ces balises aient finalement été diffusées ou non. Pour calculer le taux d'achèvement de la vidéo (magnétoscope), divisez la somme de <code>Avail.Complete</code> par la somme de <code>Avail.Impression</code> . Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>Avail.ObservedDuration</code>	Nombre total observé de millisecondes de diffusion d'annonces survenues au cours de la CloudWatch période. <code>Avail.ObservedDuration</code> est émis à la fin de la diffusion de l'annonce et est basé sur la durée des segments indiqués dans le manifeste pendant la diffusion de l'annonce. Dimensions : <code>ConfigurationName</code>
<code>Avail.ObservedFilledDuration</code>	Le nombre observé de millisecondes de temps de disponibilité des publicités qui ont été MediaTailor remplies au cours de cette période. CloudWatch Dimensions : <code>ConfigurationName</code>
<code>Avail.ObservedFillRate</code>	La moyenne simple observée des taux auxquels une annonce individuelle MediaTailor remplie est disponible au cours de la CloudWatch période. Émis uniquement pour les manifestes HLS, à la première CUE-IN balise. S'il n'y a pas de CUE-IN balise, MediaTailor n'émet pas cette métrique. Dimensions : <code>ConfigurationName</code>
<code>Avail.ObservedSlateDuration</code>	Nombre total observé de millisecondes d'ardoise insérées au cours de la CloudWatch période. Dimensions : <code>ConfigurationName</code>

Métrique	Description
GetManifest.Age	L'âge total du manifeste en millisecondes. Mesuré entre le moment où l'origine crée le manifeste et le moment où le manifeste personnalisé est MediaTailor envoyé. Pour plus d'informations sur les mesures permettant de mesurer l'âge des manifestes, consultez Utilisation de métriques pour diagnostiquer les manifestes périmés. Dimensions : ConfigurationName
GetManifest.Errors	Le nombre d'erreurs reçues lors MediaTailor de la génération des manifestes au cours de la CloudWatch période que vous avez spécifiée. Dimensions : ConfigurationName
GetManifest.Latency	Temps de MediaTailor réponse en millisecondes pour la demande de génération de manifestes. Pour plus d'informations sur les mesures permettant de mesurer l'âge des manifestes, consultez Utilisation de métriques pour diagnostiquer les manifestes périmés. Dimensions : ConfigurationName

Métrique	Description
GetManifest.MediaTailorAge	Durée pendant laquelle le manifeste a été stocké MediaTailor en millisecondes. Mesuré entre le moment où MediaTailor reçoit une réponse d'origine et le moment où le manifeste personnalisé est MediaTailor envoyé. Pour plus d'informations sur les mesures permettant de mesurer l'âge des manifestes, consultez Utilisation de métriques pour diagnostiquer les manifestes périmés. Dimensions : ConfigurationName
Origin.Age	Durée pendant laquelle l'origine possède le manifeste, en millisecondes. Mesuré entre le moment où l'origine crée le manifeste et le moment où MediaTailor la demande d'origine est envoyée. Toutes les <code>origin.*</code> métriques sont émises pour les demandes traitées directement depuis l'origine. Ils ne sont pas émis pour les réponses d'origine mises en cache. Pour plus d'informations sur les mesures permettant de mesurer l'âge des manifestes, consultez Utilisation de métriques pour diagnostiquer les manifestes périmés. Dimensions : ConfigurationName

Métrique	Description
<code>Origin.Errors</code>	Le nombre de réponses par code d'état non HTTP 200 et de réponses expirées MediaTailor reçues du serveur d'origine au cours de la CloudWatch période que vous avez spécifiée. Toutes les <code>origin.*</code> métriques sont émises pour les demandes traitées directement depuis l'origine. Ils ne sont pas émis pour les réponses d'origine mises en cache. Dimensions : <code>ConfigurationName</code>
<code>Origin.ManifestFileSizeBytes</code>	Taille du fichier du manifeste d'origine en octets pour HLS et DASH. Généralement, cette métrique est utilisée conjointement avec <code>Origin.ManifestFileSizeTooLarge</code>. Toutes les <code>origin.*</code> métriques sont émises pour les demandes traitées directement depuis l'origine. Ils ne sont pas émis pour les réponses d'origine mises en cache. Dimensions : <code>ConfigurationName</code>
<code>Origin.ManifestFileSizeTooLarge</code>	Le nombre de réponses provenant de l'origine dont la taille du manifeste est supérieure à la quantité configurée. Généralement, cette métrique est utilisée conjointement avec <code>Origin.ManifestFileSizeBytes</code>. Toutes les <code>origin.*</code> métriques sont émises pour les demandes traitées directement depuis l'origine. Ils ne sont pas émis pour les réponses d'origine mises en cache. Dimensions : <code>ConfigurationName</code>

Métrique	Description
Origin.Timeouts	Le nombre de demandes expirées adressées au serveur d'origine au cours de la CloudWatch période que vous avez spécifiée. Toutes les <code>origin.*</code> métriques sont émises pour les demandes traitées directement depuis l'origine. Ils ne sont pas émis pour les réponses d'origine mises en cache. Dimensions : <code>ConfigurationName</code>
Requests	Le nombre de transactions simultanées par seconde pour tous les types de demandes. Le nombre de transactions dépend principalement du nombre de joueurs et de la fréquence à laquelle les joueurs demandent des manifests mis à jour. Chaque demande de lecteur est comptabilisée comme une transaction. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>SkippedReason.DurationExceeded</code>	Le nombre d'annonces qui n'ont pas été insérées dans une offre car l'ADS a renvoyé une durée d'annonces supérieure à la durée de disponibilité spécifiée. Une valeur élevée pour cette métrique peut contribuer à un écart entre les <code>AdDecisionServer.Ads</code> métriques <code>AdsBilled</code> et. Pour plus d'informations sur les raisons pour lesquelles les annonces ont été ignorées, consultez Résolution des problèmes liés à l'omission d'annonces. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.EarlyCueIn</code>	Le nombre d'annonces ignorées en raison d'une annonce CUE - IN anticipée. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.ImportError</code>	Le nombre d'annonces ignorées en raison d'une erreur lors de la tâche d'importation. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>SkippedReason.ImportInProgress</code>	Le nombre d'annonces ignorées en raison d'une tâche d'importation active existante. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.InternalError</code>	Le nombre d'annonces ignorées en raison d'une erreur MediaTailor interne. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.NewCreative</code>	Le nombre d'annonces qui n'ont pas été insérées en vain car c'était la première fois que l'actif était demandé par un client. Une valeur élevée pour cette métrique peut temporairement contribuer à une baisse globale <code>Avail.FillRate</code> , jusqu'à ce que les actifs puissent être transcodés avec succès. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.NoVariantMatch</code>	Le nombre d'annonces ignorées en raison de l'absence de correspondance de variante entre l'annonce et le contenu. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Métrique	Description
<code>SkippedReason.PersonalizationThresholdExceeded</code>	Durée des publicités dépassant le seuil de personnalisation défini dans cette configuration. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.ProfileNotFound</code>	Le nombre d'annonces ignorées car le profil de transcodage n'a pas été trouvé. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.TranscodeError</code>	Le nombre de publicités ignorées en raison d'une erreur de transcodage. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)
<code>SkippedReason.TranscodeInProgress</code>	Le nombre d'annonces qui n'ont pas été insérées en vain car l'annonce n'avait pas encore été transcodée. Une valeur élevée pour cette métrique peut temporairement contribuer à une baisse globale <code>Avail.FillRate</code>, jusqu'à ce que les actifs puissent être transcodés avec succès. Dimensions : • <code>ConfigurationName</code> • Account-level (pas de dimensions)

Note

Pour les sessions HLS Interstitials, certaines métriques se comportent différemment en raison de la nature contraignante tardive des décisions publicitaires :

- `Avail.ObservedFilledDuration` correspond `Avail.FilledDuration` car il est impossible d'observer le comportement de lecture réel côté client.
- `Avail.ObservedSlateDuration` indique la durée de l'ardoise planifiée à partir des réponses de la liste des actifs plutôt que de la lecture observée.
- Les métriques préfixées par « Observé » fournissent des valeurs estimées pour les sessions HLS Interstitials.

Métriques des fonctions

Les statistiques suivantes sont publiées lorsque vous utilisez [Fonctions](#) avec vos configurations de lecture. Ces métriques sont toujours émises lorsqu'une fonction est configurée sur un hook de cycle de vie. Aucune configuration d'opt-in ou de journalisation n'est requise.

Hook-level métriques : un point de données par exécution d'un hook de cycle de vie :

Métrique	Description
<code>PreSessionInitHook.Invocations</code>	Le nombre de fois que le hook d'initialisation de pré-session a été invoqué. Dimensions : <code>ConfigurationName</code>
<code>PreSessionInitHook.Errors</code>	Le nombre d'exécutions de hooks d'initialisation de pré-session qui ont entraîné une erreur. Dimensions : <code>ConfigurationName</code>
<code>PreSessionInitHook.Latency</code>	Temps d'exécution en millisecondes du hook d'initialisation de pré-session. Dimensions : <code>ConfigurationName</code>

Métrique	Description
<code>PreAdsRequestHook.Invocations</code>	Le nombre de fois que le hook de demande préalable à la publicité a été invoqué. Dimensions : <code>ConfigurationName</code>
<code>PreAdsRequestHook.Errors</code>	Le nombre d'exécutions de hooks de requêtes antérieures à la publicité qui ont entraîné une erreur. Dimensions : <code>ConfigurationName</code>
<code>PreAdsRequestHook.Latency</code>	Temps d'exécution en millisecondes du hook de requête préalable à la publicité. Dimensions : <code>ConfigurationName</code>

Function-level métriques : un point de données par exécution de fonction individuelle. Ces mesures incluent des dimensions supplémentaires (`FunctionId`, `FunctionType`, `HookType`) qui vous permettent d'identifier quelle fonction spécifique est lente ou défectueuse :

Métrique	Description
<code>Function.Invocations</code>	Le nombre de fois qu'une fonction individuelle a été exécutée. Dimensions : <code>ConfigurationName</code> <code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>
<code>Function.Errors</code>	Le nombre d'exécutions de fonctions individuelles qui ont entraîné une erreur. Dimensions : <code>ConfigurationName</code>

Métrique	Description
	• ConfigurationName , FunctionId , FunctionType , HookType
Function.Latency	Temps d'exécution en millisecondes d'une fonction individuelle. Dimensions : • ConfigurationName • ConfigurationName , FunctionId , FunctionType , HookType

Moyennes simples et pondérées

Vous pouvez récupérer la moyenne simple et la moyenne pondérée des réponses de l'ADS aux demandes d'annonces émanant de MediaTailor et pour connaître la manière dont MediaTailor les annonces sont utilisées :

- Les moyennes simples sont fournies dans `AdDecisionServer.FillRate` et `Avail.FillRate`. Il s'agit des moyennes des pourcentages de taux de remplissage pour les diffusions de publicités individuelles pour la période. Les moyennes simples ne prennent pas en compte les différences entre les durées des diffusions de publicités individuelles.
- Les moyennes pondérées sont les pourcentages de taux de remplissage correspondant à la somme de toutes les durées de diffusion. Elles sont calculées sous la forme de $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$ et $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$. Ces moyennes reflètent les différences de durée de chaque diffusion de publicités, en privilégiant celles qui ont une durée plus longue.

Dans le cas d'une période qui contient une seule diffusion de publicités, la moyenne simple fournie par `AdDecisionServer.FillRate` est égale à la moyenne pondérée fournie par $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$. La moyenne simple fournie par `Avail.FillRate` est égale à la moyenne pondérée fournie par $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$.

Exemple

Supposons que la période que vous avez spécifiée inclut les deux diffusions de publicités suivantes :

- La première diffusion de publicités a une durée de 90 secondes :
 - La réponse ADS pour la diffusion fournit 45 secondes de publicités (50 % remplis).
 - MediaTailor remplit 45 secondes de la durée de diffusion de l'annonce (50 % de remplissage).
- La seconde diffusion de publicités a une durée de 120 secondes :
 - La réponse ADS pour la diffusion fournit 120 secondes de publicités (100 % remplis).
 - MediaTailor remplit 90 secondes de la durée de diffusion de l'annonce (75 %).

Les métriques sont les suivantes :

- `Avail.Duration` est égal à 210, la somme des deux durées de diffusion de publicités : $90 + 120$.
- `AdDecisionServer.Duration` est égal à 165, la somme des deux durées de réponse : $45 + 120$.
- `Avail.FilledDuration` est égal à 135, la somme des deux durées remplies : $45 + 90$.
- `AdDecisionServer.FillRate` est égal à 75 %, la moyenne des pourcentages remplis pour chaque diffusion : $(50 \% + 100 \%) / 2$. Il s'agit de la moyenne simple.
- La moyenne pondérée pour les taux de remplissage ADS est égale à 78,57 %, ce qui correspond à `AdDecisionServer.Duration` en tant que pourcentage de `Avail.Duration` : $(165 * 100) / 210$. Ce calcul tient compte des différences de durées.
- `Avail.FillRate` est égal à 62,5 %, la moyenne des pourcentages remplis pour chaque diffusion : $(50 \% + 75 \%) / 2$. Il s'agit de la moyenne simple.
- La moyenne pondérée des taux MediaTailor de remplissage disponibles est de 64,29 %, soit un `Avail.FilledDuration` pourcentage de `Avail.Duration` : $(135 * 100) / 210$. Ce calcul tient compte des différences de durées.

Le maximum `Avail.FillRate` que l' MediaTailor on peut atteindre pour toute annonce disponible est de 100 %. L'ADS peut renvoyer plus temps de publicité que ce qui est disponible dans la diffusion, mais MediaTailor peut remplir seulement le temps disponible.

AWS Elemental MediaTailor CloudWatch dimensions

Vous pouvez filtrer les AWS Elemental MediaTailor données à l'aide des dimensions suivantes.

Dimension	Description
ConfigurationName	Indique la configuration à laquelle la métrique appartient. Disponible pour tous les indicateurs.
FunctionId	Identifiant de la fonction. Disponible uniquement <code>Function.*</code> pour les métriques.
FunctionType	Type de fonction : <code>CUSTOM_OUTPUT</code> , <code>HTTP_REQUEST_SEQUENTIAL_EXECUTOR</code> , ou <code>CONCURRENT_EXECUTOR</code> . Disponible uniquement <code>Function.*</code> pour les métriques.
HookType	Le hook du cycle de vie qui a déclenché la fonction : <code>PRE_SESSION_INITIALIZATION</code> ou <code>PRE_ADS_REQUEST</code> . Disponible uniquement <code>Function.*</code> pour les métriques.

Utilisation de métriques pour diagnostiquer les manifestes périmés provenant de AWS Elemental MediaTailor

Un manifeste périmé est un manifeste qui n'a pas été mis à jour récemment. Les différents flux de travail d'insertion d'annonces peuvent avoir une tolérance variable quant au temps qui doit s'écouler avant qu'un manifeste ne soit considéré comme obsolète, en fonction de divers facteurs (tels que les exigences relatives aux systèmes en aval). Vous pouvez utiliser les CloudWatch métriques Amazon pour identifier les manifestes qui dépassent la tolérance d'obsolescence de votre flux de travail et aider à identifier ce qui pourrait être à l'origine des retards dans les mises à jour des manifestes.

Les indicateurs suivants permettent d'identifier les manifestes périmés et leurs causes.

Pour plus d'informations sur toutes les MediaTailor mesures émises, consultez [AWS Elemental MediaTailor CloudWatch métriques](#).

Métrique	Définition	Utilisation
GetManifest.Age	Mesure l'âge total du manifeste , y compris les deux GetManifest	Vous pouvez utiliser cette métrique pour identifier les manifestes qui

Métrique	Définition	Utilisation
	<code>st.MediaTailorAge</code> et <code>Origin.Age</code> pour cette configuration.	dépassent votre seuil de mise à jour et qui sont périmés. Définissez des alarmes sur cette métrique afin d'être alerté lorsque des manifestes périmés sont diffusés. Pour plus d'informations sur les alarmes, consultez la section Alarme sur les métriques dans le Guide de CloudWatch l'utilisateur Amazon. Lorsque vous recevez une alarme, <code>Origin.Age</code> utilisez-la <code>GetManifest</code> <code>st.MediaTailorAge</code> pour identifier si MediaTailor ou si l'origine est à l'origine de l'obsolescence.
<code>Origin.Age</code>	Mesure la durée pendant laquelle l'origine possède le manifeste avant de l'envoyer MediaTailor pour cette configuration. Cette métrique n'est pas émise lorsque la réponse provient d'un réseau de diffusion de contenu (CDN). La réponse doit provenir de l'origine <code>Origin.Age</code> pour être émise.	Lorsque vous identifiez des manifestes périmés avec <code>GetManifest.Age</code>, vous pouvez analyser la <code>Origin.Age</code> métrique et la <code>GetManifest.MediaTailorAge</code> métrique pour déterminer ce qui contribue à l'obsolescence des manifestes. Si vous constatez que cela <code>Origin.Age</code> est plus long que vos temps de traitement habituels à l'origine, il est probable que le problème soit dû au système en amont et vous devez vous concentrer sur ce point pour les diagnostics.

Métrique	Définition	Utilisation
<code>GetManifest.MediaTailorAge</code>	Mesure la durée MediaTailor de stockage de ce manifeste pour cette configuration.	Lorsque vous identifiez des manifestes périmés avec <code>GetManifest.Age</code>, vous pouvez analyser la <code>GetManifest.MediaTailorAge</code> métrique et la <code>Origin.Age</code> métrique pour déterminer ce qui contribue à l'obsolescence des manifestes. Si le délai de personnalisation de votre manifeste <code>GetManifest.MediaTailorAge</code> est plus long que d'habitude MediaTailor, il MediaTailor est probable que cela soit à l'origine du problème et vous devez y concentrer les diagnostics. <code>GetManifest.Latency</code> peut mieux identifier le temps nécessaire MediaTailor pour créer un manifeste personnalisé.

Métrique	Définition	Utilisation
GetManifest.Latency	Mesure le temps nécessaire MediaTailor au traitement de la demande et à la création d'un manifeste personnalisé pour cette configuration.	Lorsque vous comparez <code>Origin.Age</code> et <code>GetManifest.MediaTailorAge</code> et déterminez que MediaTailor c'est la cause du retard de livraison du manifeste, vous pouvez analyser la <code>GetManifest.Latency</code> métrique pour déterminer si le processus de personnalisation du manifeste contribue à l'obsolescence du manifeste. <code>GetManifest.MediaTailorAge</code> mesure la durée totale pendant laquelle le manifeste est stocké MediaTailor. <code>GetManifest.Latency</code> mesure la durée de stockage consacrée à la MediaTailor personnalisation du manifeste en réponse à une demande.

Enregistrement AWS Elemental MediaTailor Appels d'API

AWS Elemental MediaTailor est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans MediaTailor. CloudTrail capture tous les appels d'API en MediaTailor tant qu'événements. Les appels capturés incluent les appels depuis la MediaTailor console et les appels de code vers les opérations de l' MediaTailor API. Si vous créez un historique, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris des événements pour MediaTailor. Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents sur la CloudTrail console dans l'historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite MediaTailor, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des informations supplémentaires.

Pour en savoir plus CloudTrail, consultez le Guide de [AWS CloudTrail l'utilisateur](#).

AWS Elemental MediaTailor informations dans CloudTrail

CloudTrail est activé sur votre AWS compte lorsque vous le créez. Lorsqu'une activité se produit AWS Elemental MediaTailor, cette activité est enregistrée dans un CloudTrail événement avec les autres événements de AWS service dans l'historique des événements. Vous pouvez consulter, rechercher et télécharger les événements récents sur votre AWS compte. Pour plus d'informations, consultez la section [Affichage des événements avec l'historique des CloudTrail événements](#).

Pour un enregistrement permanent des événements de votre AWS compte, y compris les événements pour AWS Elemental MediaTailor, créez un historique. Un journal permet CloudTrail de transmettre des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un historique dans la console, celui-ci s'applique à toutes les AWS régions. Le journal enregistre les événements de toutes les régions de la AWS partition et transmet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en profondeur les données d'événements collectées dans les CloudTrail journaux et y donner suite. Pour plus d'informations, consultez les ressources suivantes :

- [Création d'un historique pour votre AWS compte](#)
- [AWS intégrations de services avec journaux CloudTrail](#)
- [Configuration des notifications Amazon SNS pour CloudTrail](#)
- [Réception de fichiers CloudTrail journaux de plusieurs régions](#) et [réception de fichiers CloudTrail journaux de plusieurs comptes](#)

Toutes les AWS Elemental MediaTailor actions sont enregistrées CloudTrail et documentées dans la référence de l'[AWS Elemental MediaTailor API](#). Par exemple, les appels aux opérations PutPlaybackConfiguration et ListPlaybackConfigurations génèrent des entrées dans les fichiers journaux CloudTrail.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été faite avec l'utilisateur root ou les informations d' Gestion des identités et des accès AWS identification (IAM)
- Si la demande a été effectuée avec des informations d'identification de sécurité temporaires pour un rôle ou un utilisateur fédéré

- Si la demande a été faite par un autre AWS service

Pour de plus amples informations, veuillez consulter [Élément CloudTrail userIdentity](#).

Compréhension AWS Elemental MediaTailor entrées de fichier journal

Un journal est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment Amazon S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande individuelle à partir d'une source quelconque et comprend des informations sur l'action demandée, sur tous les paramètres, les paramètres de la demande, etc. Les fichiers journaux CloudTrail ne sont pas des séries ordonnées retraçant les appels d'API publics. Ils ne suivent aucun ordre précis.

L'exemple suivant montre une entrée de CloudTrail journal qui illustre l'PutPlaybackConfigurationaction :

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:53:46Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "PutPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIpAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "examplename",
    "AdDecisionServerUrl": "http://exampleads.com"
  },
  "responseElements": {
    "SessionInitializationEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/session/AKIAIOSFODNN7EXAMPLE/examplename/",
    "DashConfiguration": {
```

```

      "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/AKIAIOSFODNN7EXAMPLE/exemplename/",
      "MpdLocation": "EMT_DEFAULT"
    },
    "AdDecisionServerUrl": "http://exampleads.com",
    "CdnConfiguration": {},
    "PlaybackEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com",
    "HlsConfiguration": {
      "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/AKIAIOSFODNN7EXAMPLE/exemplename/"
    },
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "exemplename"
  },
  "requestID": "1a2b3c4d-1234-5678-1234-1a2b3c4d5e6f",
  "eventID": "987abc65-1a2b-3c4d-5d6e-987abc654def",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}

```

L'exemple suivant montre une entrée de CloudTrail journal qui illustre l'GetPlaybackConfigurationaction :

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:52:37Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "GetPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "Name": "exemplename"
  }
}

```

```

},
"responseElements": null,
"requestID": "0z1y2x3w-0123-4567-9876-6q7r8s9t0u1v",
"eventID": "888ddd77-3322-eeew-uuii-abc123jkl343",
"readOnly": true,
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}

```

Réception AWS Elemental MediaTailor alertes d'assemblage de canaux

MediaTailor crée des alertes en cas de problèmes réels ou potentiels liés à vos ressources d'assemblage de canaux. L'alerte décrit le problème, la date à laquelle il s'est produit et les ressources concernées.

Vous pouvez afficher les alertes dans les Console de gestion AWS, les kits de développement logiciel AWS Command Line Interface (AWS CLI), ou par programmation à l'aide de l'API.

MediaTailor [ListAlerts](#)

Important

Les alertes ne sont disponibles que pour les ressources d'assemblage de chaînes créées le 14 juillet 2021 ou après cette date.

Alertes d'assemblage de chaînes

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
Source de VOD	NOT_PROCESSED	MediaTailor n'a pas traité la configuration du package <i>configurationPath</i> .	
	UNREACHABLE	Impossible d'accéder à l'URL <i>url</i> .	

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
	UNAUTHORIZED	<i>url</i> n'a pas autorisé la demande.	
	TIMEOUT	La connexion à <i>url</i> a expiré.	
	UNPARSABLE_MANIFEST	MediaTailor a rencontré un problème lors de l'analyse du manifeste depuis <i>url</i> .	
	VARIANT_DURATION_MISMATCH	MediaTailor a rencontré des variantes dont les durées totales ne correspondaient pas lors de l'analyse du manifeste à partir de <i>url</i> . Cela peut provoquer un blocage pendant la lecture.	La durée de votre manifeste varie d'une personne à variants/représentations l'autre. Cela peut entraîner des sous-titres manquants ou incorrects et l'impossibilité d'insérer des publicités.
	SEGMENT_DURATION_TOO_LONG	MediaTailor a rencontré un segment d'une durée supérieure à trente secondes lors de l'analyse du manifeste à partir de <i>url</i> . Cela peut entraîner un blocage de la lecture, des sous-titres manquants ou incorrects et l'impossibilité d'insérer des publicités.	Votre manifeste contient un segment de plus de 30 secondes.

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
	TARGET_DURATION_MISMATCH	MediaTailor a rencontré un décalage de EXT-X-TARGETDURATION valeurs entre les manifestes HLS lors de l'analyse du manifeste à partir de. <i>url</i> Cela peut provoquer un blocage pendant la lecture.	La durée cible ne correspond pas à tous les manifestes de la source.
Emplacement de la source	NOT_PROCESSED	MediaTailor n'a pas traité la ressource <i>resourceName</i> .	
Programme	VOD_SOURCE_ALERT	La source VOD <i>vodSourceName</i> de ce programme comporte l'alerte suivante : <i>vodSourceAlertCode</i> <i>vodSourceAlertMessage</i>	
	SOURCE_LOCATION_ALERT	L'emplacement source <i>sourceLocationName</i> contenu dans ce programme comporte l'alerte suivante <i>sourceLocationAlertCode</i> : <i>sourceLocationAlertMessage</i>	

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
	CODEC_MISMATCH	MediaTailor J'ai rencontré un codec incompatible dans <i>channelName</i> le calendrier. L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	
	RESOLUTION_MISMATCH	MediaTailor a rencontré une résolution non concordante dans le <i>channelName</i> calendrier. L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	
	BANDWIDTH_MISMATCH	MediaTailor a rencontré une incompatibilité de bande passante dans le <i>channelName</i> calendrier. L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
	FRAMERATE_MISMATCH	MediaTailor a rencontré une fréquence d'images non adaptée dans le calendrier. <i>channelName</i> L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	
	TARGET_DURATION_MISMATCH	MediaTailor a rencontré des EXT-X-TARGETDURATION valeurs incompatibles entre les manifestes HLS dans le <i>channelName</i> calendrier. L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	
	SEGMENT_DURATION_MISMATCH	MediaTailor a rencontré des valeurs de durée de segment incompatibles entre les manifestes du <i>channelName</i> calendrier. L'inadéquation se situe <i>sourceGroupName</i> entre <i>programName1</i> le manifeste <i>manifestUrl</i> et <i>programName2</i> le manifeste <i>manifestUrl</i> .	

Type d'alerte	Code d'alerte	Message d'alerte	Remarques
	NO_COMMON_SEGMENT_BOUNDARY_FOR_AD_SLATE	MediaTailor Je n'ai pas pu insérer de liste d'annonces en décalage <i>offsetMillis</i> pour le programme <i>programName</i> . Il n'existe pas de limite de segment commune à l'heure de début de la liste d'annonces.	
	NOT_PROCESSED	MediaTailor n'a pas traité la ressource <i>resourceName</i> .	
	TOO_MANY_ALERTS	MediaTailor a détecté trop d'alertes et n'en fournira plus pour <i>programName</i> . Effacez les alertes existantes pour continuer à recevoir des alertes <i>programName</i> .	
Canal	PROGRAM_ALERT	Le programme <i>programName</i> contenu dans cette chaîne comporte l'alerte suivante <i>programAlertCode</i> : <i>programAlertMessage</i>	

Résolution des problèmes liés aux alertes liées aux manifestes

Lorsque vous recevez des alertes liées à un manifeste `UNPARSABLE_MANIFEST`, telles que, ou `VARIANT_DURATION_MISMATCH` `TARGET_DURATION_MISMATCH`, vérifiez les points suivants :

1. Vérifiez les exigences relatives à la source : assurez-vous que votre contenu répond aux exigences décrites dans [the section called “Intégrer une source de contenu”](#).
2. Consultez les journaux des manifestes : pour une analyse détaillée des erreurs de manifeste et les types d'événements, consultez [the section called “Journaux des manifestes”](#).

3. Vérifiez les codes d'erreur : pour une analyse détaillée des erreurs, consultez [Résolution des problèmes](#).
4. Vérifiez le format du manifeste : vérifiez que votre manifeste respecte les spécifications de format appropriées pour votre protocole de diffusion (HLS ou DASH).

Affichage des alertes

Vous pouvez consulter les alertes pour n'importe quelle ressource d'assemblage de MediaTailor canaux. Lorsque vous consultez les alertes relatives aux chaînes et aux programmes, MediaTailor toutes les ressources connexes contenues dans la chaîne ou le programme sont incluses. Par exemple, lorsque vous consultez les alertes relatives à un programme spécifique, vous voyez également des alertes concernant l'emplacement de la source et les sources de VOD que contient le programme.

Pour consulter les alertes, effectuez la procédure suivante.

Console

Pour afficher les alertes dans la console

1. Ouvrez la MediaTailor console à l'adresse <https://console.aws.amazon.com/mediatailor/>.
2. Choisissez la ressource pour laquelle vous souhaitez afficher les alertes.
3. Sélectionnez l'onglet **Alertes** pour afficher les alertes.

AWS Command Line Interface (AWS CLI)

Pour répertorier les alertes relatives à une ressource d'assemblage de canaux, vous avez besoin du nom de ressource [Amazon \(ARN\) de la ressource](#). Vous pouvez utiliser la `describe-resource_type` commande contenue dans le AWS Command Line Interface (AWS CLI) pour obtenir l'ARN de la ressource. Par exemple, exécutez la [commande](#) `describe-channel` pour obtenir l'ARN d'un canal spécifique :

```
aws mediatailor describe-channel --channel-name MyChannelName
```

Utilisez ensuite la [commande](#) `aws mediatailor list-alerts` pour répertorier les alertes associées à la ressource :

```
aws mediatailor list-alerts --resource-arn arn:aws:mediatailor:region:aws-account-id:resource-type/resource-name
```

API

Pour répertorier les alertes relatives à une ressource d'assemblage de canaux, vous avez besoin du nom de ressource [Amazon \(ARN\) de la ressource](#). Vous pouvez utiliser l'`DescribeResource` opération dans l' API MediaTailor pour obtenir l'ARN de la ressource. Par exemple, utilisez l'`DescribeChannel` opération pour obtenir l'ARN d'un canal spécifique.

Utilisez ensuite l'`ListAlerts` API pour répertorier les alertes relatives à la ressource.

Gestion des alertes

Lorsqu'une alerte se produit, consultez les alertes dans les Console de gestion AWS AWS SDK ou utilisez les AWS Command Line Interface (AWS CLI), ou l'API MediaTailor Alerts pour déterminer les sources possibles du problème.

Une fois le problème résolu, MediaTailor efface l'alerte.

Identification AWS Elemental MediaTailor resources

Une balise est une étiquette de métadonnées que vous attribuez ou que vous AWS attribuez à une AWS ressource. Chaque balise se compose d'une clé et d'une valeur. Pour les balises que vous attribuez, vous définissez la clé et la valeur. Par exemple, vous pouvez définir la clé sur `stage` et la valeur pour une ressource sur `test`.

Les balises vous permettent d'effectuer les actions suivantes :

- Identifiez et organisez vos AWS ressources. De nombreux AWS services prennent en charge le balisage. Vous pouvez donc attribuer le même tag à des ressources provenant de différents services pour indiquer que les ressources sont liées. Par exemple, vous pouvez attribuer la même étiquette à un AWS Elemental MediaPackage canal et à un point de terminaison que vous attribuez à une AWS Elemental MediaTailor configuration.
- Suivez vos AWS coûts. Vous activez ces balises sur le AWS Billing and Cost Management tableau de bord. AWS utilise les balises pour classer vos coûts et vous fournir un rapport mensuel de répartition des coûts. Pour plus d'informations, consultez [Utilisation des balises de répartition des coûts](#) dans le [Guide de l'utilisateur AWS Billing](#).

- Contrôlez l'accès à vos AWS ressources. Pour de plus amples informations, veuillez consulter [Contrôle de l'accès à l'aide d'identifications](#) dans le [Guide de l'utilisateur IAM](#).

Les sections suivantes fournissent plus d'informations sur les balises pour AWS Elemental MediaTailor.

Ressources prises en charge dans AWS Elemental MediaTailor

Les ressources suivantes dans AWS Elemental MediaTailor Supports Balisting :

- Canaux
- Configurations
- SourceLocations
- VodSources

Restrictions liées aux balises

Les restrictions de base suivantes s'appliquent aux balises des AWS Elemental MediaTailor ressources :

- Nombre maximum d'étiquettes que vous pouvez attribuer à une ressource – 50
- Longueur de clé maximale : 128 caractères Unicode
- Longueur de valeur maximale : 256 caractères Unicode
- Caractères valides pour la clé et la valeur : a-z A-Z, 0-9, espace, et les caractères suivants : _ . : / = + - et @
- Les clés et les valeurs distinguent les majuscules et minuscules
- N'utilisez pas aws : comme préfixe pour les clés ; seul AWS peut utiliser cette valeur

Gestion des balises dans AWS Elemental MediaTailor

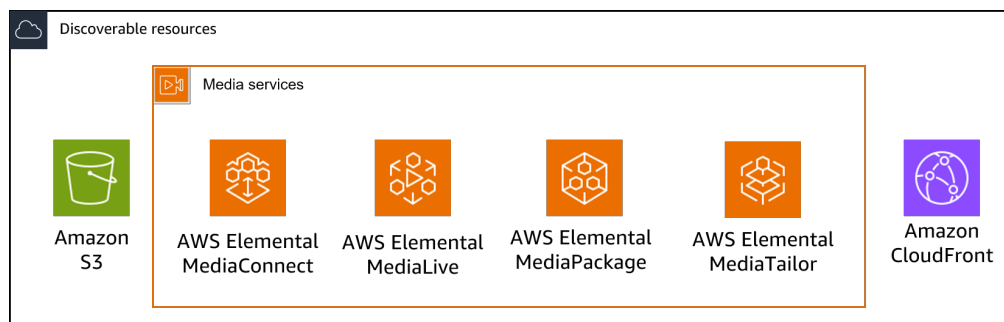
Vous définissez des balises en tant que propriétés sur une ressource. Vous pouvez ajouter, modifier et supprimer des balises via l' AWS Elemental MediaTailor API ou le AWS Command Line Interface (AWS CLI). Pour plus d'informations, consultez la [Référence des API AWS Elemental MediaTailor](#).

Contrôle AWS services multimédia avec moniteur de flux de travail

Le moniteur de flux de travail est un outil permettant de découvrir, de visualiser et de surveiller AWS les flux de travail multimédia. Le moniteur de flux de travail est disponible dans la AWS console et l'API. Vous pouvez utiliser le moniteur de flux de travail pour découvrir et créer des mappages visuels des ressources de votre flux de travail, appelés cartes de signaux. Vous pouvez créer et gérer des modèles CloudWatch d'alarme Amazon et de EventBridge règles Amazon pour surveiller les ressources mappées. Les modèles de surveillance que vous créez sont transformés en AWS CloudFormation modèles déployables pour permettre la répétabilité. AWS les modèles d'alarme recommandés fournissent une surveillance des meilleures pratiques prédéfinies.

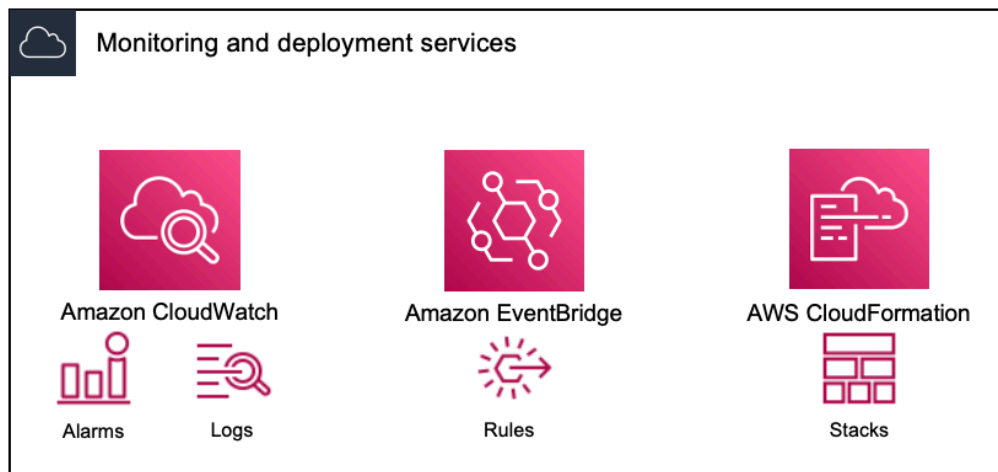
Découvrez

Utilisez des cartes de signaux pour découvrir automatiquement les AWS ressources interconnectées associées à votre flux de travail multimédia. La découverte peut commencer à partir de n'importe quelle ressource de service prise en charge et créer un mappage de bout en bout du flux de travail. Les cartes de signaux peuvent être utilisées comme outils de visualisation autonomes ou améliorées avec des modèles de surveillance.



Surveiller

Vous pouvez créer des modèles CloudWatch d'alarmes et de EventBridge règles personnalisés pour surveiller la santé et l'état de vos flux de travail multimédia. Des modèles d'alarme conformes aux meilleures pratiques sont disponibles et peuvent être importés dans votre environnement de surveillance des flux de travail. Vous pouvez utiliser les modèles d'alarme conformes aux meilleures pratiques tels quels ou les modifier pour mieux les adapter à votre flux de travail. Tous les modèles que vous créez sont transformés en CloudFormation modèles pour un déploiement reproductible.



Note

L'utilisation du moniteur de flux de travail n'entraîne aucun coût direct. Cependant, des coûts sont associés aux ressources créées et utilisées pour surveiller votre flux de travail.

Lorsque la surveillance est déployée, Amazon CloudWatch et les EventBridge ressources Amazon sont créées. Lorsque vous utilisez la console de AWS gestion, avant de déployer la surveillance sur une carte de signal, vous serez informé du nombre de ressources qui seront créées. Pour plus d'informations sur la tarification, voir : [CloudWatch tarification](#) et [EventBridge tarification](#).

Le moniteur de flux de travail utilise des AWS CloudFormation modèles pour déployer les EventBridge ressources CloudWatch et. Ces modèles sont stockés dans un compartiment Amazon Simple Storage Service de classe standard qui est créé pour vous, par Workflow Monitor, pendant le processus de déploiement et entraînera des frais de stockage et de rappel des objets. Pour plus d'informations sur la tarification, consultez : Tarification [Amazon S3](#).

Les aperçus générés dans la carte des signaux du moniteur de flux de travail pour les AWS Elemental MediaPackage chaînes sont fournis depuis le point de terminaison MediaPackage Origin et entraînent des frais de transfert de données sortantes. Pour la tarification, voir : [MediaPackage tarification](#).

Composants du moniteur de flux de travail

Le moniteur de flux de travail comporte quatre composants principaux :

- CloudWatch modèles d'alarme : définissez les conditions que vous souhaitez surveiller CloudWatch. Vous pouvez créer vos propres modèles d'alarme ou importer des modèles prédéfinis créés par AWS. Pour plus d'informations, voir : [CloudWatch groupes d'alarmes et modèles pour surveiller votre AWS flux de travail multimédia](#)
- EventBridge modèles de règles : définissez la manière dont les notifications sont EventBridge envoyées lorsqu'une alarme est déclenchée. Pour plus d'informations, voir : [EventBridge groupes de règles et modèles pour surveiller votre AWS flux de travail multimédia](#)
- Cartes des signaux : utilisez un processus automatisé pour créer des cartes de flux de travail AWS Elemental à l'aide des AWS ressources existantes. Les cartes de signaux peuvent être utilisées pour découvrir les ressources de votre flux de travail et déployer la surveillance de ces ressources. Pour plus d'informations, voir : [Cartes des signaux du moniteur de flux de travail](#)
- Vue d'ensemble - La page d'aperçu vous permet de surveiller directement l'état de plusieurs cartes de signaux à partir d'un seul endroit. Passez en revue les statistiques, les journaux et les alarmes de vos flux de travail. Pour plus d'informations, voir : [Présentation du moniteur de flux de travail](#)

Services pris en charge

Le moniteur de flux de travail prend en charge la découverte automatique et le mappage des signaux des ressources associées aux services suivants :

- AWS Elemental MediaConnect
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaTailor
- Amazon S3
- Amazon CloudFront

Rubriques

- [Configuration du moniteur de flux de travail pour surveiller AWS services multimédias](#)
- [Utilisation du moniteur de flux de travail](#)

Configuration du moniteur de flux de travail pour surveiller AWS services multimédias

Pour configurer le moniteur de flux de travail pour la première fois, vous créez les modèles d'alarme et d'événement et vous découvrez les cartes de signaux utilisées pour surveiller vos flux de travail multimédia. Le guide suivant contient les étapes nécessaires pour configurer les rôles IAM de niveau administrateur et opérateur, créer des ressources de surveillance des flux de travail et déployer la surveillance sur vos flux de travail.

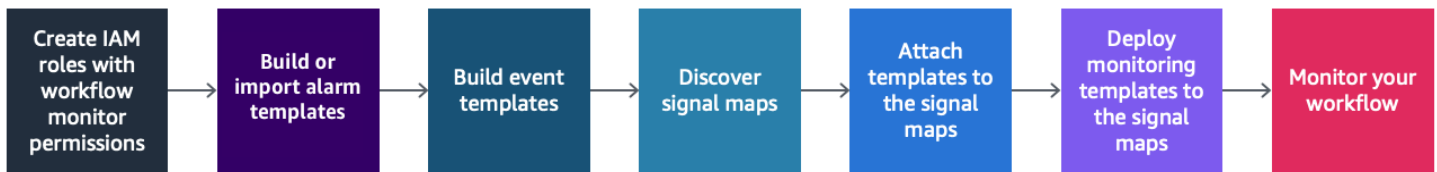
Rubriques

- [Premiers pas avec Workflow Monitor](#)
- [Groupes et modèles de moniteurs de flux de travail](#)
- [Cartes des signaux du moniteur de flux de travail](#)
- [Quotas de surveillance des flux](#)

Premiers pas avec Workflow Monitor

Les étapes suivantes fournissent un aperçu de base de l'utilisation de Workflow Monitor pour la première fois.

1. Configurez les autorisations IAM du moniteur de flux de travail pour les rôles de niveau administrateur et opérateur : [Politiques IAM de surveillance des flux de travail](#)
2. Créez des modèles d'alarme ou importez des modèles prédéfinis créés par AWS : [Alarmes CloudWatch](#)
3. Créez des événements de notification qui seront envoyés par EventBridge : [EventBridge règles](#)
4. Découvrez des cartes de signaux à l'aide de vos ressources AWS élémentaires existantes : [Cartes des signaux](#)
5. Joignez les modèles d'alarme et les règles de notification à votre carte des signaux : [Joindre des modèles](#)
6. Déployez les modèles pour commencer à surveiller la carte des signaux : [Déploiement de modèles de surveillance](#)
7. Surveillez et passez en revue les ressources de surveillance de votre flux de travail à l'aide de la section de présentation de la AWS console : [Présentation de](#)



Politiques IAM de surveillance des flux de travail

Le moniteur de flux de travail interagit avec plusieurs AWS services pour créer des cartes de signaux, créer EventBridge des ressources CloudWatch et des CloudFormation modèles. Étant donné que le moniteur de flux de travail interagit avec un large éventail de services, des politiques Gestion des identités et des accès AWS (IAM) spécifiques doivent être attribuées à ces services. Les exemples suivants indiquent les politiques IAM nécessaires pour les rôles IAM d'administrateur et d'opérateur.

Politique IAM de l'administrateur

L'exemple de politique suivant concerne une politique IAM de surveillance des flux de travail au niveau de l'administrateur. Ce rôle permet de créer et de gérer des ressources de surveillance des flux de travail et des ressources de service prises en charge qui interagissent avec le moniteur des flux de travail.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:List*",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:PutAnomalyDetector",
        "cloudwatch:PutMetricData",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:PutCompositeAlarm",
        "cloudwatch:PutDashboard",
        "cloudwatch>DeleteAlarms",
        "cloudwatch>DeleteAnomalyDetector",
        "cloudwatch>DeleteDashboards",
        "cloudwatch:TagResource",

```

```
        "cloudwatch:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:List*",
        "cloudformation:Describe*",
        "cloudformation:CreateStack",
        "cloudformation:UpdateStack",
        "cloudformation>DeleteStack",
        "cloudformation:TagResource",
        "cloudformation:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudfront:List*",
        "cloudfront:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "events:List*",
        "events:Describe*",
        "events:CreateEventBus",
        "events:PutRule",
        "events:PutTargets",
        "events:EnableRule",
        "events:DisableRule",
        "events>DeleteRule",
        "events:RemoveTargets",
```

```
        "events:TagResource",
        "events:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*",
        "logs:TagLogGroup",
        "logs:TagResource",
        "logs:UntagLogGroup",
        "logs:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "medialive:*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackagev2:List*",
```

```
        "mediapackagev2:Get*",
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage-vod:List*",
        "mediapackage-vod:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediatailor:List*",
        "mediatailor:Describe*",
        "mediatailor:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "resource-groups:ListGroup",
        "resource-groups:GetGroup",
        "resource-groups:GetTags",
        "resource-groups:GetGroupQuery",
        "resource-groups:GetGroupConfiguration",
        "resource-groups:CreateGroup",
        "resource-groups:UngroupResources",
        "resource-groups:GroupResources",
        "resource-groups>DeleteGroup",
        "resource-groups:UpdateGroupQuery",
        "resource-groups:UpdateGroup",
        "resource-groups:Tag",
        "resource-groups:Untag"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "s3:"
```

```

    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "sns:TagResource",
      "sns:UntagResource"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "tag:Get*",
      "tag:Describe*",
      "tag:TagResources",
      "tag:UntagResources"
    ],
    "Resource": "*"
  }
]
}

```

Politique IAM de l'opérateur

L'exemple de politique suivant concerne une politique IAM de surveillance des flux de travail au niveau de l'opérateur. Ce rôle permet un accès limité et en lecture seule aux ressources du moniteur de flux de travail et aux ressources de service prises en charge qui interagissent avec le moniteur de flux de travail.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:List*",
        "cloudwatch:Describe*",

```



```
        "cloudwatch:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:List*",
        "cloudformation:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudfront:List*",
        "cloudfront:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "events:List*",
        "events:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*"
    ],
    "Resource": "*"
},
{
```

```
    "Effect": "Allow",
    "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "medialive:List*",
        "medialive:Get*",
        "medialive:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackagev2:List*",
        "mediapackagev2:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage-vod:List*",
        "mediapackage-vod:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediatailor:List*",
```

```

        "mediatailor:Describe*",
        "mediatailor:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "s3:Get*",
        "s3:List*"
    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
},
{
    "Effect": "Allow",
    "Action": [
        "tag:Get*",
        "tag:Describe*"
    ],
    "Resource": "*"
}
]
}

```

Groupes et modèles de moniteurs de flux de travail

Avant de pouvoir déployer la surveillance du flux de travail sur une carte des signaux, vous devez créer les groupes et les modèles pour les CloudWatch alarmes et EventBridge les notifications. Les CloudWatch modèles définissent les scénarios et les seuils qui seront utilisés pour déclencher les alarmes. Les EventBridge modèles détermineront la manière dont ces alarmes vous sont signalées.

Si vous souhaitez uniquement des mappages de vos ressources connectées et que vous ne souhaitez pas utiliser les fonctionnalités de modèle de surveillance de Workflow Monitor, les cartes de signaux peuvent être utilisées sans EventBridge modèles CloudWatch et. Pour plus d'informations sur l'utilisation des cartes de signaux, consultez : [Cartes des signaux](#)

Rubriques

- [CloudWatch groupes d'alarmes et modèles pour surveiller votre AWS flux de travail multimédia](#)
- [EventBridge groupes de règles et modèles pour surveiller votre AWS flux de travail multimédia](#)

CloudWatch groupes d'alarmes et modèles pour surveiller votre AWS flux de travail multimédia

Les alarmes du moniteur de flux de travail vous permettent d'utiliser CloudWatch les mesures existantes comme base des alarmes pour vos cartes de signaux. Vous pouvez créer un groupe de modèles d'alarme pour trier et classer les types d'alarmes importants pour votre flux de travail. Dans chaque groupe de modèles d'alarme, vous créez des modèles d'alarme avec CloudWatch des mesures et des paramètres spécifiques que vous souhaitez surveiller. Vous pouvez créer vos propres modèles d'alarme ou importer des modèles d'alarme recommandés créés par AWS. Après avoir créé un groupe de modèles d'alarme et des modèles d'alarme au sein de ce groupe, vous pouvez associer un ou plusieurs de ces groupes de modèles d'alarme à une carte de signaux.

Vous devez d'abord créer un groupe de modèles d'alarme. Après avoir créé un groupe de modèles d'alarme, vous pouvez créer vos propres modèles ou utiliser les modèles recommandés créés par AWS. Si vous souhaitez créer vos propres modèles d'alarme, continuez sur cette page. Pour plus d'informations sur l'importation de modèles recommandés, consultez : [Modèles recommandés](#)

Cette section traite de la création d' CloudWatch alarmes à l'aide de Workflow Monitor. Pour plus d'informations sur la manière dont le CloudWatch service gère les alarmes et pour plus de détails sur les composants des alarmes, voir : [Utilisation CloudWatch des alarmes](#) dans le guide de CloudWatch l'utilisateur Amazon

Création de groupes de modèles d'alarme

Les groupes de modèles d'alarmes vous permettent de trier et de classer les types d'alarmes qui sont importants pour votre flux de travail.

Pour créer un groupe de modèles d'alarme

1. Dans le volet de navigation de la console Workflow Monitor, sélectionnez des modèles CloudWatch d'alarme.
2. Sélectionnez Créer un groupe de modèles d'alarme.
3. Attribuez au groupe de modèles d'alarme un nom de groupe unique et une description facultative.
4. Sélectionnez Créer, vous serez redirigé vers la page de détails du groupe de modèles d'alarme nouvellement créé.

Création de modèles d'alarme

Vous pouvez créer des modèles d'alarme avec les CloudWatch mesures et les paramètres que vous souhaitez surveiller.

Pour créer un modèle d'alarme

1. Sur la page de détails du groupe de modèles d'alarme, sélectionnez **Créer un modèle d'alarme**.
2. Attribuez au modèle d'alarme un nom de modèle unique et une description facultative.
3. Dans la section **Choisir une métrique** :
 1. Sélectionnez un type de ressource cible. Le type de ressource cible est une ressource pour le service concerné, tel qu'un canal pour MediaLive MediaPackage et/ou un flux pour MediaConnect.
 2. Sélectionnez un nom de métrique. Il s'agit de la CloudWatch métrique qui sert de base à l'alarme. La liste des mesures changera en fonction du type de ressource cible sélectionné.
4. Dans la section **Paramètres de l'alarme** :

Note

Pour plus d'informations sur la manière dont le CloudWatch service gère les alarmes et pour plus de détails sur les composants des alarmes, voir : [Utilisation CloudWatch des alarmes](#) dans le guide de CloudWatch l'utilisateur Amazon

1. Sélectionnez la statistique. Il s'agit d'une valeur telle qu'une somme ou une moyenne qui sera utilisée pour surveiller la métrique.
2. Sélectionnez l'opérateur de comparaison. Ce champ fait référence au seuil que vous avez défini à l'étape suivante.
3. Définissez un seuil. Il s'agit d'une valeur numérique que l'opérateur de comparaison utilise pour déterminer un statut supérieur, inférieur ou égal au statut.
4. Définissez une période. Il s'agit d'une valeur temporelle, en secondes. La période est la durée pendant laquelle la statistique, l'opérateur de comparaison et le seuil interagissent pour déterminer si l'alarme est déclenchée.
5. Définissez les points de données. Cette valeur détermine le nombre de points de données nécessaires pour déclencher l'alarme.

6. Sélectionnez le mode de traitement des données manquantes. Cette sélection détermine la façon dont cette alarme réagit aux données manquantes.
5. Sélectionnez **Créer** pour terminer le processus.

Un exemple de modèle d'alarme complet peut comporter les paramètres suivants : Un type de ressource cible de MediaConnect flux est surveillé pour le nom de la métrique de déconnexions. La valeur statistique est définie sur Somme avec un opérateur de comparaison « supérieur ou égal à » et un seuil de 10. La période est fixée à 60 secondes et ne nécessite qu'un point de données sur 1. Traiter les données manquantes est réglé sur « Ignorer ».

Le résultat de ces paramètres est le suivant : le moniteur de flux de travail surveillera les déconnexions sur le flux. Si 10 déconnexions ou plus se produisent dans les 60 secondes, l'alarme sera déclenchée. 10 déconnexions ou plus en 60 secondes ne doivent se produire qu'une seule fois pour que l'alarme soit déclenchée.

Modèles d'alarme recommandés pour surveiller votre AWS flux de travail multimédia

Les modèles recommandés par Workflow Monitor sont une sélection organisée de mesures de service AWS Elemental avec des paramètres d'alarme prédéfinis adaptés à la métrique. Si vous ne souhaitez pas créer de modèles d'alarme personnalisés, les modèles recommandés vous fournissent des modèles de surveillance conformes aux meilleures pratiques créés par AWS.

Le moniteur de flux de travail contient des groupes de modèles recommandés pour chaque service pris en charge. Ces groupes sont conçus pour appliquer la surveillance des meilleures pratiques à des types spécifiques de flux de travail. Chaque groupe de modèles contient une sélection organisée d'alarmes configurées à partir de mesures spécifiques au service. Par exemple, un groupe de modèles recommandé pour un flux de travail MediaLive multiplex comportera un ensemble de mesures préconfigurées différent de celui d'un flux de travail MediaConnect CDI.

Pour utiliser les modèles d'alarme recommandés

1. Suivez les étapes pour [créer un groupe](#) de modèles d'alarme ou sélectionnez-en un existant.
2. Dans la section Modèles d'alarme, sélectionnez **Importer**. Vous devrez importer les modèles AWS recommandés dans votre groupe de modèles.
3. Utilisez la liste déroulante des groupes de modèles d'CloudWatch alarme pour sélectionner un groupe AWS recommandé. Ces groupes contiennent des alarmes organisées pour des services spécifiques.

4. Sélectionnez les modèles à importer à l'aide des cases à cocher. Chaque modèle répertorie ses métriques, ses valeurs de surveillance préconfigurées et fournit une description de la métrique. Lorsque vous avez terminé de sélectionner les modèles, cliquez sur le bouton **Ajouter**.
5. Les modèles sélectionnés seront déplacés vers la section **Modèles d'alarme à importer**. Vérifiez vos choix et sélectionnez **Importer**.
6. Une fois l'importation terminée, les modèles sélectionnés seront ajoutés au groupe de modèles. Si vous souhaitez ajouter d'autres modèles, répétez le processus d'importation.
7. Les modèles importés peuvent être personnalisés après l'importation. Les paramètres d'alarme peuvent être modifiés pour répondre à vos besoins en matière d'alarme.

EventBridge groupes de règles et modèles pour surveiller votre AWS flux de travail multimédia

CloudWatch utilise les EventBridge règles d'Amazon pour envoyer des notifications. Vous commencez par créer un groupe de modèles d'événements. Dans ce groupe de modèles d'événements, vous créez des modèles d'événements qui déterminent quelles conditions créent une notification et qui est averti.

Cette section traite de la création de EventBridge règles à l'aide de Workflow Monitor. Pour plus d'informations sur la manière dont le EventBridge service utilise les règles, voir : [EventBridge règles](#) dans le guide de EventBridge l'utilisateur Amazon

Création de groupes de modèles d'événements

Les groupes de modèles d'événements vous permettent de trier et de classer les événements en fonction de votre cas d'utilisation.

Pour créer un groupe de modèles d'événements

1. Dans le volet de navigation de la console Workflow Monitor, sélectionnez des modèles de EventBridge règles.
2. Sélectionnez **Créer un groupe de modèles d'événements**.
3. Donnez au groupe de modèles d'alarme un nom de groupe unique et une description facultative.
4. Sélectionnez **Créer**, vous serez redirigé vers la page de détails du groupe de modèles d'alarme nouvellement créé.

Création de modèles d'événements

Vous pouvez envoyer des notifications en fonction des modèles d'événements que vous créez.

Pour créer un modèle d'événement

1. Sur la page de détails du groupe de modèles d'événements, sélectionnez **Créer un modèle d'événement**.
2. Donnez au modèle d'événement un nom de modèle unique et une description facultative.
3. Dans la section **Paramètres des règles** :
 1. Sélectionnez un type d'événement. Lorsque vous sélectionnez un type d'événement, vous pouvez choisir entre plusieurs événements créés par AWS ou sélectionner **Signal map active alarm** pour utiliser une alarme créée par un modèle d'alarme.
 2. Sélectionnez un service **Target**. Cela détermine la manière dont vous souhaitez être informé de cet événement. Vous pouvez sélectionner **Amazon Simple Notification Service** ou **CloudWatch les journaux**.
 3. Après avoir sélectionné un service cible, sélectionnez un service cible. Il s'agira d'une rubrique **Amazon SNS** ou d'un groupe de **CloudWatch journaux**, selon le service que vous avez sélectionné.
4. Sélectionnez **Créer** pour terminer le processus.

Cartes des signaux du moniteur de flux de travail

Les cartes de signaux sont des mappages visuels des AWS ressources de votre flux de travail multimédia. Vous pouvez utiliser le moniteur de flux de travail pour démarrer la découverte de la carte des signaux sur l'un des types de ressources pris en charge. Au cours du processus de découverte, le moniteur de flux de travail cartographiera automatiquement et récursivement toutes les AWS ressources connectées. Une fois la carte des signaux créée, vous pouvez utiliser la console de surveillance des flux de travail pour déployer des modèles de surveillance, afficher des mesures et afficher les détails des ressources mappées.

Rubriques

- [Création de cartes de signaux pour AWS flux de travail multimédia](#)
- [Affichage des cartes des signaux de AWS flux de travail multimédia](#)
- [Joindre des modèles d'alarme et d'événement à la carte des signaux de votre AWS flux de travail multimédia](#)

- [Déploiement de modèles sur la carte des signaux de votre AWS flux de travail multimédia](#)
- [Mise à jour de la carte des signaux de votre AWS flux de travail multimédia](#)
- [Supprimer la carte des signaux de votre AWS flux de travail multimédia](#)

Création de cartes de signaux pour AWS flux de travail multimédia

Vous pouvez utiliser les cartes des signaux du moniteur de flux de travail pour créer une cartographie visuelle de toutes les AWS ressources connectées dans votre flux de travail multimédia.

Pour créer une carte des signaux

1. Dans le volet de navigation de la console Workflow Monitor, sélectionnez **Signal maps**.
2. Sélectionnez **Créer une carte des signaux**.
3. Donnez un nom et une description à la carte des signaux.
4. Dans la section **Découvrez une nouvelle carte de signal**, les ressources du compte actuel et de la région sélectionnée sont affichées. Sélectionnez une ressource pour commencer la découverte de la carte des signaux. La ressource sélectionnée sera le point de départ de la découverte.
5. Sélectionnez **Créer**. Patientez quelques instants pour terminer le processus de découverte. Une fois le processus terminé, la nouvelle carte des signaux vous sera présentée.

Note

Les aperçus générés dans la carte des signaux du moniteur de flux de travail pour les AWS Elemental MediaPackage chaînes sont fournis depuis le point de terminaison MediaPackage Origin et entraînent des frais de transfert de données sortantes. Pour la tarification, voir : [MediaPackage tarification](#).

Affichage des cartes des signaux de AWS flux de travail multimédia

Les cartes des signaux du moniteur de flux de travail vous permettent de visualiser une cartographie visuelle de toutes les AWS ressources connectées dans votre flux de travail multimédia.

Vues cartographiques des signaux

Après avoir sélectionné une carte des signaux, vous disposez de deux vues qui peuvent être utilisées pour surveiller ou configurer la carte des signaux. Surveiller la carte des signaux et configurer

la carte des signaux est un bouton contextuel situé en haut à droite de la section console de cartographie des signaux.

Si vous sélectionnez la carte des signaux à l'aide de la section Cartes des signaux du volet de navigation, votre carte des signaux sera affichée dans la vue de configuration. La vue de configuration vous permet d'apporter des modifications aux groupes de modèles associés à cette carte de signal, de déployer les modèles joints et d'afficher les informations de base et les balises de la carte de signal.

Si vous sélectionnez la carte des signaux à l'aide de la section Vue d'ensemble du volet de navigation, votre carte des signaux sera affichée en mode surveillance. La vue de surveillance affiche les CloudWatch alarmes, EventBridge les règles, les alertes, les journaux et les mesures pour cette carte des signaux.

La vue peut être modifiée à tout moment en sélectionnant le bouton de la carte des Monitor/ Configure signaux en haut à droite. La vue de configuration nécessite des autorisations IAM de niveau administrateur. Les autorisations IAM requises peuvent être consultées ici : [Politiques IAM de surveillance des flux de travail](#)

Naviguer sur la carte des signaux

Une carte des signaux contiendra des nœuds pour chaque ressource AWS prise en charge découverte par Workflow Monitor. Certaines ressources, telles que les MediaLive chaînes et les MediaPackage points de terminaison, peuvent afficher des aperçus miniatures du contenu, si des aperçus miniatures sont disponibles.

En sélectionnant un nœud de ressource, puis en sélectionnant Afficher les détails des ressources sélectionnées dans le menu déroulant Actions, vous serez redirigé vers la page de détails du service associé. Par exemple, la sélection d'une MediaLive chaîne et la sélection de Afficher les détails des ressources sélectionnées ouvriront la page de détails de la MediaLive console pour cette chaîne.

La sélection d'un nœud de ressource permet de filtrer la liste des alarmes actives uniquement pour ce nœud. Si vous sélectionnez l'ARN cible de la ressource dans l'alarme active, vous serez redirigé vers la page de détails du service associé, avec la ressource sélectionnée ouverte.

Joindre des modèles d'alarme et d'événement à la carte des signaux de votre AWS flux de travail multimédia

Après avoir créé des modèles d'alarme et d'événement, vous devez les joindre à une carte des signaux. Tous les modèles d'alarmes et d'événements que vous avez créés peuvent être joints à toutes les cartes de signaux découvertes.

Pour joindre des modèles d'alarme et d'événement à votre carte des signaux

1. Dans le volet de navigation de la console de surveillance des flux de travail, sélectionnez **Cartes des signaux**, puis sélectionnez la carte des signaux avec laquelle vous souhaitez travailler.
2. En haut à droite de la page de carte des signaux, dans l'onglet **Groupe de modèles CloudWatch d'alarme**, sélectionnez **Joindre des groupes de modèles CloudWatch d'alarme**.
 1. Dans la nouvelle section qui s'ouvre, choisissez tous les groupes de modèles d'alarme que vous souhaitez appliquer à cette carte de signaux, puis sélectionnez **Ajouter**. Cela entraînera le déplacement des groupes de modèles d'alarme sélectionnés vers la section **Groupe de modèles CloudWatch d'alarme joints**.
 2. Sélectionnez **Enregistrer pour** enregistrer vos modifications et revenir à la page de la carte des signaux.
3. À droite de la page de la carte des signaux, sélectionnez l'onglet **Groupe de modèles de EventBridge règles**, puis sélectionnez **Joindre des groupes de modèles de EventBridge règles**.
 1. Dans la nouvelle section qui s'ouvre, choisissez tous les groupes de modèles d'événements que vous souhaitez appliquer à cette carte de signaux, puis sélectionnez **Ajouter**. Les groupes de modèles de règles sélectionnés seront alors déplacés vers la section **Groupe de modèles de EventBridge règles joints**.
 2. Sélectionnez **Enregistrer pour** enregistrer vos modifications et revenir à la page de la carte des signaux.
4. Vous avez attribué des modèles CloudWatch d'alarme et de EventBridge règle à la carte des signaux, mais la surveillance n'est pas encore déployée. La section suivante portera sur le déploiement des ressources de surveillance.

Déploiement de modèles sur la carte des signaux de votre AWS flux de travail multimédia

Après avoir joint les modèles d'alarme et d'événement à votre carte des signaux, vous devez déployer la surveillance. Tant que le déploiement n'est pas terminé, la surveillance de votre carte des signaux ne sera pas active.

Le moniteur de flux de travail ne déploiera que les alarmes pertinentes pour la carte de signal sélectionnée. Par exemple, le groupe de modèles d'alarme joint peut contenir des alarmes pour plusieurs services, tels que MediaLive MediaPackage, et MediaConnect. Si la carte de signal sélectionnée ne contient que des MediaLive ressources, aucune MediaConnect alarme MediaPackage ou aucune alarme ne sera déployée.

Pour déployer les modèles de surveillance

1. Après avoir associé des groupes de modèles d'alarmes et d'événements à votre carte de signaux et enregistré vos modifications, sélectionnez **Déployer le moniteur** dans le menu déroulant **Actions**.
2. Il vous sera demandé de confirmer le déploiement et on vous présentera le nombre CloudWatch et EventBridge les ressources qui seront créées. Si vous souhaitez continuer, sélectionnez **Déployer**.

Note

L'utilisation du moniteur de flux de travail n'entraîne aucun coût direct. Cependant, des coûts sont associés aux ressources créées et utilisées pour surveiller votre flux de travail.

Lorsque la surveillance est déployée, Amazon CloudWatch et les EventBridge ressources Amazon sont créées. Lorsque vous utilisez la console de AWS gestion, avant de déployer la surveillance sur une carte de signal, vous serez informé du nombre de ressources qui seront créées. Pour plus d'informations sur la tarification, voir : [CloudWatch tarification](#) et [EventBridge tarification](#).

Le moniteur de flux de travail utilise des AWS CloudFormation modèles pour déployer les EventBridge ressources CloudWatch et. Ces modèles sont stockés dans un compartiment Amazon Simple Storage Service de classe standard qui est créé pour vous, par Workflow Monitor, pendant le processus de déploiement et entraînera des frais de stockage et de rappel des objets. Pour plus d'informations sur la tarification, consultez : Tarification [Amazon S3](#).

3. L'état du déploiement est affiché à côté du nom de la carte des signaux. L'état du déploiement est également visible dans la section **Stacks** de la CloudFormation console. Après quelques instants de création et de déploiement des ressources, la surveillance de votre carte des signaux commencera.

Mise à jour de la carte des signaux de votre AWS flux de travail multimédia

Si une modification est apportée à votre flux de travail, vous devrez peut-être redécouvrir la carte des signaux et redéployer les ressources de surveillance. Le moniteur de flux de travail est un outil de visualisation et de surveillance qui ne permet pas d'apporter des modifications à votre flux de travail. Les cartes de signaux représentent une visualisation instantanée de votre flux de travail. Si vous ajoutez, supprimez ou modifiez de manière significative des parties de votre flux de travail multimédia, nous vous recommandons de redécouvrir la carte des signaux. Si des ressources de surveillance sont associées à la carte des signaux, nous vous recommandons de redéployer la surveillance après le processus de redécouverte.

Pour redécouvrir une carte des signaux

1. Dans le volet de navigation de la console de surveillance des flux de travail, sélectionnez **Cartes des signaux**, puis sélectionnez la carte des signaux avec laquelle vous souhaitez travailler.
2. Vérifiez que vous êtes dans la vue **Configurer la carte des signaux**. Pour plus d'informations sur la modification des vues, consultez : [Affichage des cartes de signaux](#)
3. En haut à droite de la page de la carte des signaux, sélectionnez le menu déroulant **Actions**. Sélectionnez **Redécouvrir**.
4. L'écran de redécouverte vous sera présenté. Sélectionnez une ressource qui fait partie du flux de travail que vous êtes en train de redécouvrir. Sélectionnez le bouton **Redécouvrir**.
5. La carte des signaux sera reconstruite en fonction du flux de travail actuel. Si vous devez redéployer des ressources de surveillance, restez sur la page de cette carte des signaux. Tous les modèles de surveillance précédemment joints resteront joints, mais devront être redéployés.

Pour redéployer des modèles de surveillance après la redécouverte d'une carte de signaux

1. Après la redécouverte, vous serez dirigé vers la carte des signaux mise à jour. Pour redéployer les modèles de surveillance, sélectionnez **Déployer le moniteur** dans le menu déroulant **Actions**.

2. Il vous sera demandé de confirmer le déploiement et le nombre de EventBridge ressources qui seront créées vous seront présentés. CloudWatch Si vous souhaitez continuer, sélectionnez Déployer.
3. L'état du déploiement est affiché à côté du nom de la carte des signaux. Après quelques instants de création et de déploiement des ressources, la surveillance de votre carte des signaux commencera.

Supprimer la carte des signaux de votre AWS flux de travail multimédia

Si vous n'avez plus besoin d'une carte des signaux, vous pouvez la supprimer. Si des modèles de surveillance sont déployés sur la carte des signaux, le processus de suppression vous demandera de supprimer toutes CloudWatch les EventBridge ressources qui ont été déployées sur cette carte des signaux. La suppression des ressources déployées n'affecte pas les modèles qui les ont créées. Cette suppression de ressources permet de s'assurer que vous ne disposez CloudWatch pas de EventBridge ressources déployées mais non utilisées.

Pour supprimer une carte de signaux

1. Dans le volet de navigation de la console de surveillance des flux de travail, sélectionnez Cartes des signaux et sélectionnez le bouton radio à côté de la carte des signaux que vous souhaitez supprimer.
2. Sélectionnez le bouton Supprimer. Il vous sera demandé de confirmer la suppression des ressources de surveillance. Sélectionnez Supprimer pour démarrer le processus de suppression des ressources de surveillance.
3. La colonne Surveiller le déploiement affiche l'état actuel. Lorsque le statut est passé à DELETE_COMPLETE, sélectionnez à nouveau le bouton Supprimer.
4. Il vous sera demandé de confirmer la suppression de la carte des signaux. Sélectionnez Supprimer pour continuer et supprimer la carte des signaux.

Quotas de surveillance des flux

La section suivante contient des quotas pour les ressources du moniteur de flux de travail. Chaque quota est calculé « par compte ». Si vous devez augmenter un quota pour votre compte, vous pouvez utiliser la console [AWS Service Quotas](#) pour demander une augmentation, sauf indication contraire dans le tableau suivant.

Quotas

Type de ressource	Quota
CloudWatch groupes de modèles d'alarme	20
CloudWatch modèles d'alarme	200
EventBridge groupes de modèles de règles	20
EventBridge modèles de règles	200
Cartes des signaux	30
Cartes de signaux : groupes de modèles CloudWatch d'alarme associés à une seule carte de signaux	5 Vous ne pouvez pas augmenter ce quota.
Cartes de signaux : groupes de modèles de EventBridge règles associés à une seule carte de signaux	5 Vous ne pouvez pas augmenter ce quota.

Utilisation du moniteur de flux de travail

Utilisez les sections **Vue d'ensemble** et **cartes des signaux** de la console de surveillance des flux de travail pour consulter l'état actuel des flux de travail et toutes les alarmes, mesures et journaux associés.

Rubriques

- [Présentation du moniteur de flux de travail](#)
- [Vue d'ensemble des journaux et des mesures pour le suivi des flux de travail](#)
- [Utilisation des cartes de signaux du moniteur de flux de travail](#)

Présentation du moniteur de flux de travail

La section **Présentation** de la console de surveillance des flux de travail est un tableau de bord qui fournit des informations en un coup d'œil sur vos cartes de signaux. Dans la section de présentation, vous pouvez voir l'état actuel de la surveillance de chaque carte de signal, ainsi que CloudWatch les

métriques et les CloudWatch journaux associés. Vous pouvez sélectionner n'importe quelle carte de signal à rediriger vers cette page de console de cartes de signaux.

Vue d'ensemble du filtrage

À l'aide de la barre de recherche de la section d'aperçu, vous pouvez filtrer la liste des cartes de signaux à l'aide de contraintes contextuelles. Après avoir sélectionné la barre de recherche, une liste de propriétés à filtrer vous sera présentée. La sélection d'une propriété affichera des opérateurs tels que Equal, Contains, Does not equal et Does not contains. La sélection d'un opérateur créera une liste de ressources à partir du type de propriété sélectionné. Si vous sélectionnez l'une de ces ressources, la liste des cartes de signaux n'affichera que les cartes de signaux qui correspondent à la contrainte que vous avez définie.

Vue d'ensemble des journaux et des mesures pour le suivi des flux de travail

Pour afficher CloudWatch les statistiques et les journaux d'une carte des signaux, sélectionnez le bouton radio à côté du nom de la carte des signaux. Une interface à onglets pour les métriques et les journaux apparaîtra sous la liste des cartes des signaux.

CloudWatch Métriques

CloudWatch les mesures pour la carte de signal sélectionnée seront contextuelles et afficheront uniquement les mesures associées aux services utilisés dans ce flux de travail de cartes de signaux. Vous pouvez utiliser les outils de statistiques à l'écran pour personnaliser les périodes métriques et les plages de temps affichées.

CloudWatch Journaux

Si vous avez associé un groupe de CloudWatch journaux à la carte des signaux, ce groupe sera affiché ici.

Utilisation des cartes de signaux du moniteur de flux de travail

Dans la section d'aperçu de la console, vous pouvez sélectionner une carte de signal spécifique pour afficher plus d'informations sur cette carte de signal et les ressources de surveillance associées.

Après avoir sélectionné une carte des signaux, la carte des signaux et un certain nombre de sections à onglets contenant plus d'informations vous seront présentées :

- CloudWatch alarmes
- EventBridge règles

- AWS Alertes élémentaires
- Métriques
- Journaux
- Informations de base

Naviguer sur la carte des signaux

Une carte des signaux contiendra des nœuds pour chaque ressource AWS prise en charge découverte par Workflow Monitor. Certaines ressources, telles que les MediaLive chaînes et les MediaPackage terminaux, peuvent afficher des aperçus miniatures du contenu, si des aperçus miniatures sont disponibles.

En sélectionnant un nœud de ressource, puis en sélectionnant **Afficher les détails** des ressources sélectionnées dans le menu déroulant **Actions**, vous serez redirigé vers la page de détails du service associé. Par exemple, la sélection d'une MediaLive chaîne et la sélection de **Afficher les détails des ressources sélectionnées** ouvriront la page de détails de la MediaLive console pour cette chaîne.

La sélection d'un nœud de ressource permet de filtrer la liste des alarmes actives uniquement pour ce nœud. Si vous sélectionnez l'ARN cible de la ressource dans l'alarme active, vous serez redirigé vers la page de détails du service associé, avec la ressource sélectionnée ouverte.

Dual-stack points de terminaison (IPv6) pour AWS Elemental MediaTailor

AWS Elemental MediaTailor prend en charge les points de terminaison à double pile que vous pouvez utiliser pour vous connecter via IPv4 ou IPv6. Ces points de terminaison sont fonctionnellement équivalents aux points de terminaison standard. IPv4-only Le comportement du service est identique et seule la couche de transport réseau est différente.

Que sont les points de terminaison à double pile ?

Dual-stack les points de terminaison publient à la fois des adresses IPv4 et IPv6. Lorsque vous utilisez un point de terminaison à double pile, le DNS est résolu en adresse IPv4 ou IPv6 en fonction de la configuration de votre réseau et des préférences du client. Vous n'avez pas besoin de modifier vos MediaTailor ressources, vos configurations de lecture ou votre comportement d'insertion de publicités.

Fonctionnalités prises en charge

Le tableau suivant répertorie les MediaTailor fonctionnalités prises en charge pour les points de terminaison à double pile.

Fonctionnalité	Dual-stack soutien
API du plan de contrôle (toutes les régions)	Pris en charge
Plan de données SSAI (initialisation de session, manifeste, segment)	Pris en charge
Plan de données d'assemblage des canaux	Pris en charge
Console de gestion AWS	Pris en charge
AWS SDK	Prochainement

Dual-stack format du point final

MediaTailor les points de terminaison à double pile utilisent le `.api.aws` domaine :

- Plan de contrôle : `mediatailor.region.api.aws`
- Plan de données SSAI : affiché dans la section Points de terminaison Dualstack des détails de votre configuration de lecture dans le Console de gestion AWS
- Plan de données d'assemblage des canaux : affiché dans les détails de sortie de votre canal dans le Console de gestion AWS.

Vos applications existantes continuent de fonctionner avec les IPv4-only points de terminaison d'origine (`api.mediatailor.region.amazonaws.com` pour le plan de contrôle, `unique-id.mediatailor.region.amazonaws.com` pour le plan de données). Pour une liste complète des points de terminaison disponibles, consultez la section [MediaTailorPoints de terminaison et quotas](#) dans la référence AWS générale.

Utilisation des points de terminaison Dual-Stack

Console de gestion AWS

Dual-stack les points de terminaison apparaissent sur la page de détails de configuration de Playback, dans la section Points de terminaison Dualstack. Vous pouvez copier ces URL directement pour les utiliser dans la configuration de votre lecteur ou de votre CDN.

AWS Command Line Interface

Spécifiez l'URL du point de terminaison à double pile lorsque vous effectuez des appels d'API :

```
aws mediatailor get-playback-configuration \  
  --name my-config \  
  --endpoint-url https://mediatailor.us-east-1.api.aws
```

my-config Remplacez-le par le nom de votre configuration de lecture et *us-east-1* par votre AWS région.

AWS Kit SDK

Activez le double stack dans la configuration de votre SDK.

Python (boto3) :

```
import boto3

client = boto3.client(
    'mediatailor',
    endpoint_url='https://mediatailor.us-east-1.api.aws'
)
```

JavaScript (AWS SDK v3) :

```
import { MediaTailorClient } from "@aws-sdk/client-mediatailor";

const client = new MediaTailorClient({
    useDualstackEndpoint: true
});
```

Considérations

Gardez les points suivants à l'esprit lorsque vous utilisez des points de terminaison à double pile :

- Dual-stack les points de terminaison sont disponibles partout Régions AWS où ils MediaTailor sont disponibles.
- Il n'est pas nécessaire de modifier les configurations de lecture, les paramètres du serveur de décision publicitaire ou la configuration du CDN pour utiliser des points de terminaison à double pile.
- Si votre CDN utilise un routage basé sur le modèle de nom d'hôte, vérifiez que vos règles s'adaptent au format de domaine. `.api.aws`
- Les AWS directives standard pour les points de terminaison à double pile s'appliquent. Pour plus d'informations, voir [Dual-stack et points de terminaison FIPS](#) dans le manuel de référence AWS général.

Quotas en AWS Elemental MediaTailor

MediaTailor les demandes de ressources et d'opérations sont soumises aux quotas suivants (anciennement appelés « limites »).

Vous pouvez utiliser le AWS service Quotas de service pour consulter les quotas et demander des augmentations de quotas MediaTailor, ainsi que pour de nombreux autres AWS services. Pour plus d'informations, consultez le [Guide de l'utilisateur Service Quotas](#).

Quotas relatifs à l'insertion d'annonces

Le tableau suivant décrit les quotas d'insertion d' AWS Elemental MediaTailor annonces. Sauf indication contraire, les quotas ne sont pas ajustables.

Nom	Valeur du quota par défaut	Description	
Longueur du serveur ADS	25 000	Nombre maximal de caractères dans une spécification de serveur ADS.	
Redirections du serveur ADS	7	La profondeur maximale des redirections que MediaTailor suivent dans les balises d'emballage VAST. MediaTailor abandonne s'il y a des redirections supplémentaires.	
Délai d'expiration du serveur ADS	3 secondes (par défaut)	Le nombre maximum de secondes qui s'écoulent avant l'expiration	

Nom	Valeur du quota par défaut	Description	
		d'une connexion ouverte à un serveur de décision publicitaire (ADS). Lorsqu'une connexion expire en raison de l'absence de réponse de l'ADS, MediaTailor il est impossible de remplir l'annonce avec des publicités. Ce délai est configurable via le paramètre de délai d'expiration des requêtes ADS individuelles (plage : 250 à 6 000 ms). Pour de plus amples informations, veuillez consulter Réglages avancés .	

Nom	Valeur du quota par défaut	Description	
Demandes d'insertion d'annonces	10 000	Nombre maximum de demandes par seconde pour créer des manifestes personnalisés lors de l'insertion d'annonces côté serveur. L'insertion d'annonces gère les demandes entrantes de manifestes, d'initialisation de session, de données de suivi et de segments publicitaires. Ce quota est réglable.	
Configurations	1 000	Le nombre maximum de configurations MediaTailor autorisées.	
Longueur de l'origine du contenu	512	Nombre maximal de caractères dans une spécification d'origine de contenu.	

Nom	Valeur du quota par défaut	Description	
Délai d'expiration du serveur d'origine de contenu	2	Le nombre maximum de secondes qui s'écoulent avant l'expiration d'une connexion ouverte au serveur d'origine du contenu lors de la demande de modèles de manifestes. Les délais d'expiration génèrent les erreurs de réponse HTTP 504 (GatewayTimeoutException) .	
Taille du manifeste	2	Taille maximale, en Mo, de tout manifeste de lecture d'origine . Pour vous assurer de rester sous le quota, utilisez gzip pour compresser vos manifestes d'entrée dans MediaTailor.	
Configurations des packages	5	Le nombre maximum de configurations de packages par source (qu'il s'agisse de live ou de vidéo à la demande).	

Nom	Valeur du quota par défaut	Description	
Précharger les horaires	25	Nombre maximum de programmes de prélecture actifs par configuration de lecture. Les programmes de prélecture expirés ne sont pas pris en compte dans cette limite.	

Nom	Valeur du quota par défaut	Description	
Délai d'expiration de la demande de balise de signalement côté serveur	3 secondes	Le nombre maximum de secondes qui MediaTailor s'écoulent avant l'expiration d'une connexion ouverte au serveur lors du déclenchement d'une balise pour les rapports côté serveur. Lorsqu'une connexion expire, MediaTailor ne peut pas déclencher la balise et le service enregistre un <code>ERROR_FIRING_BEACON_FAILED</code> message lors de la MediaTailor/AdDecisionServerInteractions connexion CloudWatch.	

Nom	Valeur du quota par défaut	Description	
Délai d'expiration du hook PRE_SESSION_INITIALIZATION	2 secondes	Le nombre maximum de secondes que peut prendre l'exécution d'un hook PRE_SESSION_INITIALIZATION, y compris toutes les étapes de la fonction et les appels HTTP. Si le hook dépasse son délai d'expiration, MediaTailor supprime toutes les sorties et procède comme si aucune fonction n'avait été exécutée.	
Délai d'expiration du hook PRE_AD_REQUEST	2 secondes	Le nombre maximum de secondes que peut prendre l'exécution d'un hook PRE_AD_REQUEST, y compris toutes les étapes de la fonction et les appels HTTP. Si le hook dépasse son délai d'expiration, MediaTailor supprime toutes les sorties et procède comme si aucune fonction n'avait été exécutée.	

Nom	Valeur du quota par défaut	Description	
Fonctions maximales par hameçon	20	Le nombre maximum d'exécutions de fonctions autorisées par hook.	
Expiration de session	10 fois la durée du manifeste	Durée maximale pendant laquelle MediaTailor une session reste inactive avant de la mettre fin à la session. L'activité de la session peut être une demande de lecteur ou une avance par le serveur d'origine . Lorsque la session expire, MediaTailor renvoie une erreur de HTTP 400 (Bad Request) réponse.	

Quotas relatifs à l'assemblage des chaînes

Le tableau suivant décrit les quotas relatifs à l'assemblage des AWS Elemental MediaTailor canaux. Sauf indication contraire, les quotas sont [ajustables](#).

Nom	Valeur du quota par défaut	Description	
Demandes de manifeste de chaîne, par compte	400	Nombre maximum de demandes de manifeste de sortie	

Nom	Valeur du quota par défaut	Description	
		par seconde pour tous les canaux Channel Assembly d'un compte.	
Demandes de manifeste de chaîne, par chaîne	50	Le nombre maximum de demandes de manifeste de sortie par seconde pour un canal d'assemblage de canaux unique.	
Sorties de canaux	5	Le nombre maximum de sorties par canal.	
Chaînes par compte	100	Le nombre maximum de chaînes par compte.	
Sources en direct	50	Le nombre maximum de sources en direct pour l'emplacement de la source.	
Programmes par chaîne	400	Le nombre maximum de programmes par canal.	
Configurations de distribution par segment	5	Le nombre maximum de configurations de diffusion de segments par emplacement source.	

Nom	Valeur du quota par défaut	Description	
Emplacements des sources	50	Le nombre maximum d'emplacements sources par compte.	
Sources de VOD	1 000	Le nombre maximum de sources de vidéo à la demande (VOD) pour l'emplacement de la source.	

Le tableau suivant décrit les limites d'étranglement lors de l'assemblage des AWS Elemental MediaTailor canaux. Sauf indication contraire, les quotas sont [ajustables](#).

Nom	Limite maximale de transactions par seconde par défaut	Description	
ConfigureLogsForChannel	1	Configurez les journaux pour la chaîne.	
CreateChannel	1	Créez une chaîne.	
CreateLiveSource	1	Créez une source en direct.	
CreateProgram	3	Créez un programme.	
CreateSourceLocation	1	Créez un emplacement source.	
CreateVodSource	1	Créez une source de VOD.	

Nom	Limite maximale de transactions par seconde par défaut	Description	
DeleteChannel	1	Supprimez une chaîne.	
DeleteChannelPolicy	1	Supprimez une politique de chaîne.	
DeleteLiveSource	1	Supprimez une source en direct.	
DeleteProgram	3	Supprimez un programme.	
DeleteSourceLocation	1	Supprimez un emplacement source.	
DeleteVodSource	1	Supprimez une source de VOD.	
DescribeChannel	5	Décrivez une chaîne.	
DescribeLiveSource	5	Décrivez une source en direct.	
DescribeProgram	5	Décrivez un programme.	
DescribeSourceLocation	5	Décrivez un emplacement source.	
DescribeVodSource	5	Décrivez une source de VOD.	
GetChannelPolicy	5	Obtenez une politique de chaîne.	

Nom	Limite maximale de transactions par seconde par défaut	Description	
GetChannelSchedule	5	Obtenez le calendrier des chaînes.	
ListAlerts	5	Répertoriez les alertes.	
ListChannels	5	Répertoriez les chaînes.	
ListLiveSources	5	Répertoriez les sources en direct.	
ListPrograms	5	Répertoriez les programmes.	
ListSourceLocations	5	Répertoriez les emplacements des sources.	
ListTagsForResource	5	Répertoriez les balises d'une ressource.	
ListVodSources	5	Répertoriez les sources de VOD.	
PutChannelPolicy	3	Définissez une politique de chaîne.	
StartChannel	1	Démarrez une chaîne.	
StopChannel	1	Arrêtez une chaîne.	
TagResource	1	Balise une ressource.	

Nom	Limite maximale de transactions par seconde par défaut	Description	
UntagResource	1	Supprime la balise d'une ressource.	
UpdateChannel	1	Mettez à jour une chaîne.	
UpdateLiveSource	1	Mettez à jour une source en direct.	
UpdateProgram	1	Mettez à jour un programme.	
UpdateSourceLocation	1	Mettez à jour un emplacement source.	
UpdateVodSource	1	Mettez à jour une source de VOD.	

Résolution des problèmes MediaTailor

Si vous rencontrez des erreurs de lecture ou des difficultés similaires lors de l' AWS Elemental MediaTailor utilisation, consultez les rubriques de cette section.

Rubriques

- [Résolution des problèmes de flux d' MediaTailor événements](#)
- [Résolution des problèmes de lecture depuis MediaTailor](#)
- [Déduplication des balises](#)
- [Résolution des problèmes de paramètres](#)
- [MediaTailor guide de résolution des problèmes liés à l'omission d'annonces](#)

Résolution des problèmes de flux d' MediaTailor événements

La compréhension du flux d' AWS Elemental MediaTailor événements constitue une base solide pour résoudre les problèmes d'insertion d'annonces. En analysant la séquence, le calendrier et les modèles des événements, vous pouvez rapidement identifier les problèmes et mettre en œuvre des solutions ciblées.

Cette section fournit des conseils pratiques sur l'utilisation de l'analyse des flux d'événements pour diagnostiquer les problèmes. Pour comprendre les concepts de base du flux d'événements, consultez [Flux d'événements d'insertion d'annonces](#).

Identifier les flux d'événements incomplets

Les flux d'événements incomplets se produisent lorsque la séquence d'événements attendue s'arrête avant de parvenir à une personnalisation réussie du manifeste (le processus d' MediaTailor insertion d'informations publicitaires personnalisées dans le manifeste). L'identification des ruptures de flux permet d'identifier la cause première des échecs d'insertion d'annonces.

Schémas de flux incomplets courants

Les différents points de défaillance du flux d'événements indiquent des types de problèmes spécifiques, tels que les suivants.

- Le flux s'arrête après la détection d'une opportunité publicitaire : indique des problèmes liés aux marqueurs publicitaires ou au manifeste lui-même qui MediaTailor empêchent de faire une

demande ADS. Des problèmes de connectivité, de configuration ou de temporisation ADS peuvent survenir après l'envoi de la demande ADS.

- Le flux s'arrête après une demande ADS : suggère des problèmes de réponse ADS, des problèmes d'analyse VAST, des échecs du traitement créatif, des délais ADS, des erreurs de connectivité ou des problèmes de configuration tels que des URL ADS non valides qui ne sont découvertes que lorsque la demande est faite.
- Balise de suivi manquante : peut indiquer des problèmes de configuration du suivi, des problèmes de reporting côté serveur ou des lacunes de mise en œuvre côté client.

CloudWatch requêtes pour analyse de flux incomplète

Utilisez ces requêtes Amazon CloudWatch Logs Insights pour identifier les flux d'événements incomplets. Exécutez ces requêtes sur les groupes de journaux appropriés en fonction du type d'analyse requis.

Sélection du groupe de journaux :

- MediaTailor/AdDecisionServerInteractions- À utiliser pour les requêtes analysant les interactions entre les serveurs de décision publicitaire, les opportunités publicitaires et ADS-related les échecs.
- MediaTailor/TranscodeService- À utiliser pour analyser les problèmes liés aux publicités qui n'ont pas été insérées en raison de problèmes de transcodage, d'échecs du traitement créatif ou d'autres problèmes non liés à la publicité.

Exemple identifier les opportunités publicitaires sans une personnalisation réussie des manifestes

Groupe de journaux : MediaTailor/AdDecisionServerInteractions

La requête suivante identifie les opportunités publicitaires qui n'ont pas abouti à une personnalisation réussie du manifeste :

```
fields @timestamp, eventType, avail.availId, sessionId
| filter eventType = "AD_MARKER_FOUND"
| stats count() as total_opportunities by avail.availId
| join (
  fields @timestamp, eventType, avail.availId
  | filter eventType = "FILLED_AVAIL"
  | stats count() as successful_fills by avail.availId
) on avail.availId
| where ispresent(total_opportunities) and not ispresent(successful_fills)
```

```
| sort total_opportunities desc
```

Exemple analyser les taux d'achèvement du flux d'événements

Groupe de journaux : MediaTailor/AdDecisionServerInteractions

La requête suivante analyse les taux d'achèvement des différents types d'événements :

```
fields @timestamp, eventType, avail.availId
| filter eventType in ["AD_MARKER_FOUND", "MAKING_ADS_REQUEST", "VAST_RESPONSE",
  "FILLED_AVAIL", "BEACON_FIRED"]
| stats count() by eventType, avail.availId
| sort avail.availId, eventType
```

Exemple trouver des sessions avec des événements de balise manquants

Groupe de journaux : MediaTailor/AdDecisionServerInteractions

La requête suivante identifie les sessions dont les disponibilités sont remplies mais aucun événement de balise correspondant :

```
fields @timestamp, eventType, sessionId, avail.availId
| filter eventType = "FILLED_AVAIL"
| stats count() as filled_avails by sessionId
| join (
  fields @timestamp, eventType, sessionId
  | filter eventType = "BEACON_FIRED"
  | stats count() as beacon_events by sessionId
) on sessionId
| where filled_avails > 0 and (not ispresent(beacon_events) or beacon_events = 0)
| sort filled_avails desc
```

Exemple identifier les échecs d'insertion d'annonces liés au transcodage

Groupe de journaux : MediaTailor/TranscodeService

La requête suivante identifie les problèmes de transcodage qui empêchent l'insertion d'annonces avec succès :

```
fields @timestamp, eventType, sessionId, requestId
| filter eventType in ["TRANSCODE_IN_PROGRESS", "INTERNAL_ERROR", "MISSING_VARIANTS",
  "PROFILE_NOT_FOUND"]
```

```
| stats count() as transcode_issues by eventType, sessionId  
| sort transcode_issues desc
```

Analyse des problèmes de chronométrage des événements

L'analyse de la chronologie des événements permet d'identifier les goulots d'étranglement en matière de performances et d'optimiser les flux de travail d'insertion d'annonces. Des schémas temporels inhabituels indiquent souvent des problèmes sous-jacents qui affectent l'expérience du spectateur.

Seuils de synchronisation des performances

Utilisez ces seuils temporels pour identifier les problèmes de performances potentiels.

- Durée totale du flux supérieure à 5 secondes : peut avoir un impact sur l'expérience utilisateur et peut indiquer des problèmes de performances ADS, des problèmes du serveur d'origine (tels que des délais de récupération des manifestes) ou des problèmes internes, notamment MediaTailor des problèmes d'infrastructure avec NAT Gateway, DynamoDB, EC2 ou d'autres composants du système.
- Temps de réponse ADS supérieur à 2 secondes : suggère des problèmes de performances ADS ou des problèmes de latence du réseau.
- Personnalisation des manifestes de plus d'une seconde : peut indiquer des retards dans le traitement de la création, des problèmes liés au serveur d'origine (tels que des délais de récupération des manifestes) ou des problèmes internes au MediaTailor système, notamment des contraintes d'infrastructure avec NAT Gateway, DynamoDB, EC2 ou d'autres composants.

Requêtes d'analyse chronologique

Utilisez ces requêtes pour analyser les modèles de synchronisation des événements.

Exemple mesurer la durée totale du flux d'événements

La requête suivante mesure la durée totale des flux d'événements et identifie ceux qui dépassent 5 secondes :

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["AD_MARKER_FOUND", "FILLED_AVAIL"]  
| sort @timestamp asc  
| stats min(@timestamp) as start_time, max(@timestamp) as end_time by avail.availId
```

```
| eval duration_seconds = (end_time - start_time) / 1000  
| where duration_seconds > 5
```

Exemple analyser le temps de réponse de l'ADS

La requête suivante analyse les temps de réponse de l'ADS et identifie ceux qui dépassent 2 secondes :

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["MAKING_ADS_REQUEST", "VAST_RESPONSE"]  
| sort @timestamp asc  
| stats min(@timestamp) as request_time, max(@timestamp) as response_time by  
  avail.availId  
| eval ads_response_seconds = (response_time - request_time) / 1000  
| where ads_response_seconds > 2
```

Exemple identifier la lenteur de la personnalisation des manifestes

La requête suivante identifie les processus de personnalisation des manifestes qui prennent plus d'une seconde :

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["VAST_RESPONSE", "FILLED_AVAIL"]  
| sort @timestamp asc  
| stats min(@timestamp) as response_time, max(@timestamp) as filled_time by  
  avail.availId  
| eval personalization_seconds = (filled_time - response_time) / 1000  
| where personalization_seconds > 1
```

Problèmes courants liés au flux d'événements et solutions

Cette section propose des solutions aux problèmes de flux d'événements fréquemment rencontrés, classés par type de problème et par symptômes.

Échec des requêtes du serveur Ad Decision

Symptômes : le flux d'événements s'arrête après la détection d'une opportunité publicitaire. Aucun événement de demande ADS n'a été enregistré.

Causes courantes et solutions

- Erreurs de configuration de l'URL ADS : vérifiez que l'URL ADS dans votre configuration de lecture est correcte et accessible. Dans le journal des interactions avec les publicités, vous verrez un événement de demande ADS (MAKING_ADS_REQUEST) mais aucune réponse VAST correspondante, souvent accompagnée d'un événement d'erreur `ERROR_UNKNOWN` ou d'un événement similaire.
- Problèmes de connectivité réseau : vérifiez la connectivité réseau entre MediaTailor et votre ADS, notamment les règles de pare-feu et la résolution DNS.
- SSL/TLS problèmes de certificat : assurez-vous que votre ADS utilise des certificats SSL valides provenant d'une autorité de certification fiable. Pour Google Ad Manager en particulier, vous devrez peut-être contacter [AWS Support](#) pour activer un indicateur de configuration qui accepte les certificats SSL de Google.

Requête de diagnostic

La requête suivante permet de diagnostiquer les échecs des requêtes ADS en suivant la séquence des événements :

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["AD_MARKER_FOUND", "MAKING_ADS_REQUEST", "ERROR_ADS_IO",
"ERROR_UNKNOWN_HOST"]
| sort @timestamp asc
```

Échec de réponse du serveur Ad Decision

Symptômes : les requêtes ADS aboutissent mais MediaTailor ne reçoivent pas de réponse, ou des erreurs d'analyse se produisent.

Causes courantes et solutions

- Format VAST non valide : validez vos réponses ADS VAST par rapport aux normes de spécification VAST.
- Problèmes liés au délai d'attente de l'ADS : augmentez les paramètres du délai d'attente de l'ADS ou optimisez le temps de réponse de l'AD
- Inventaire publicitaire vide : vérifiez la disponibilité de l'inventaire publicitaire et les critères de ciblage dans votre configuration ADS.

Requête de diagnostic

La requête suivante permet de diagnostiquer les échecs de réponse ADS en examinant les événements de demande et de réponse :

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["MAKING_ADS_REQUEST", "VAST_RESPONSE", "EMPTY_VAST_RESPONSE",
"ERROR_ADS_RESPONSE_PARSE", "ERROR_ADS_TIMEOUT"]
| sort @timestamp asc
```

Défaillances de personnalisation manifestes

Symptômes : des réponses VAST ont été reçues mais la personnalisation du manifeste échoue ou les publicités sont ignorées.

Causes courantes et solutions :

- Problèmes liés au transcodage créatif : vérifiez s'il s'agit d'une `NEW_CREATIVE` annonce qui nécessite un transcodage avant son insertion. Vous pouvez également vérifier les erreurs de transcodage en consultant le MediaTailor/TranscodeService journal pour détecter les événements d'erreur tels que `INTERNAL_ERROR`, `MISSING_VARIANTS`, ou `PROFILE_NOT_FOUND`.
- Problèmes d'inadéquation des durées : vérifiez que les durées des publicités correspondent aux durées d'interruption des publicités disponibles.
- Problèmes liés au seuil de personnalisation : vérifiez les paramètres du seuil de personnalisation dans votre configuration de lecture.

Requête de diagnostic

La requête suivante permet de diagnostiquer les échecs de personnalisation des manifestes en examinant les réponses VAST et les valeurs renseignées :

```
fields @timestamp, eventType, sessionId, skippedAds
| filter sessionId = "your-session-id"
| filter eventType in ["VAST_RESPONSE", "FILLED_AVAIL", "WARNING_NO_ADVERTISEMENTS"]
| sort @timestamp asc
```

Requête pour des raisons liées à l'omission d'une annonce

La requête suivante fournit des informations détaillées sur les raisons pour lesquelles les annonces ont été ignorées :

```
fields @timestamp, eventType, sessionId, skippedAds.reason, skippedAds.creativeUniqueId
| filter sessionId = "your-session-id"
| filter eventType = "WARNING_NO_ADVERTISEMENTS" or ispresent(skippedAds)
| sort @timestamp asc
```

Requête pour des raisons liées à l'absence d'annonces et à des identifiants uniques créatifs

La requête suivante fournit des informations détaillées sur les annonces ignorées, y compris les raisons et les identifiants uniques créatifs pour les deux premières annonces de chaque offre :

```
fields @timestamp, eventType
| filter sessionId = "your-session-id"
| filter eventType = "FILLED_AVAIL"
| fields avail.skippedAds.0.vastDuration as SkippedDur_Ad0,
  avail.skippedAds.0.skippedReason as Ad0_SkipReason,
  avail.skippedAds.0.creativeUniqueId as SkippedCreative0_UID
| fields avail.skippedAds.1.vastDuration as SkippedDur_Ad1,
  avail.skippedAds.1.skippedReason as Ad1_SkipReason,
  avail.skippedAds.1.creativeUniqueId as SkippedCreative1_UID
| sort @timestamp desc
```

Suivi des défaillances des balises

Symptômes : personnalisation réussie du manifeste, mais balises de suivi manquantes ou défaillantes.

Causes courantes et solutions

- Client-side problèmes de mise en œuvre : la plupart des problèmes liés aux balises de suivi proviennent de problèmes d'implémentation côté client, tels que le fait de ne pas interroger les URL de suivi assez fréquemment pour le suivi côté client, ou des problèmes de logique de déclenchement de balises spécifiques au joueur.
- Problèmes d'accessibilité des URL de suivi : vérifiez que les URL de suivi des réponses VAST sont accessibles et renvoient les réponses appropriées. Des problèmes peuvent survenir lorsque les URL ne sont pas accessibles ou lorsque MediaTailor des problèmes internes empêchent l'envoi d'une réponse de suivi réussie.

- Problèmes liés aux demandes de segments de joueurs : des défaillances apparentes des balises de suivi peuvent survenir lorsque le joueur client ne demande aucun segment. Il en résulte qu'aucune balise n'est envoyée, ce qui apparaît comme un échec de suivi mais est en fait un problème d'implémentation du joueur plutôt qu'un problème de balise.

Requête de diagnostic

La requête suivante permet de diagnostiquer les défaillances des balises de suivi en examinant les valeurs remplies et les événements des balises :

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["FILLED_AVAIL", "BEACON_FIRED", "ERROR_FIRING_BEACON_FAILED"]
| sort @timestamp asc
```

Meilleures pratiques en matière de surveillance du flux d'événements

Mettez en œuvre ces pratiques de surveillance pour identifier et résoudre de manière proactive les problèmes de flux d'événements :

Configuration des CloudWatch alarmes

Créez des CloudWatch alarmes Amazon pour surveiller les indicateurs clés du flux d'événements.

- Alarme relative au taux d'achèvement du flux : alerte lorsque le ratio de personnalisation réussie du manifeste par rapport aux opportunités publicitaires tombe en dessous de seuils acceptables.
- Alarme de temps de réponse ADS : surveillez les temps de réponse moyens des ADS et alertez lorsqu'ils dépassent les seuils de performance.
- Alarme de taux d'erreur : suivez la fréquence des événements d'erreur et alertez en cas de pics inhabituels dans des types d'erreur spécifiques.

Requêtes de surveillance régulières

Exécutez régulièrement ces requêtes pour conserver une visibilité sur l'état du flux d'événements :

Exemple taux de réussite du flux d'événements quotidien

La requête suivante fournit un aperçu quotidien des taux de réussite des flux d'événements par type d'événement :

```
fields @timestamp, eventType
| filter @timestamp > datefloor(@timestamp, 1d)
| stats count() as total_events by eventType
| sort total_events desc
```

Exemple tendance du taux d'erreur horaire

La requête suivante permet de suivre les taux d'erreur par heure afin d'identifier les problèmes courants :

```
fields @timestamp, eventType
| filter eventType like /ERROR_/
| stats count() as error_count by datefloor(@timestamp, 1h) as hour
| sort hour desc
```

Conseils pour l'optimisation des performances

Utilisez l'analyse du flux d'événements pour optimiser les performances d'insertion d'annonces.

- Optimisation des ADS : collaborez avec votre fournisseur ADS pour optimiser les temps de réponse et réduire la latence.
- Préparation créative : des créations Pre-transcode publicitaires adaptées à vos profils de contenu et réduisant les délais de traitement.
- Réglage de la configuration : ajustez les paramètres de délai d'expiration, les seuils de personnalisation et d'autres paramètres de configuration en fonction de l'analyse du flux d'événements.

Ressources supplémentaires pour la résolution des problèmes

Pour obtenir des conseils de dépannage supplémentaires allant au-delà de l'analyse du flux d'événements :

- Pour obtenir des informations détaillées sur le format de journal et les spécifications techniques, consultez [Affichage des journaux](#).
- Pour une résolution complète des problèmes courants liés à l'insertion d'annonces, consultez [Dépannage des problèmes courants](#).
- Pour obtenir des conseils sur la configuration de la surveillance et des alertes, consultez [Contrôle AWS Elemental MediaTailor avec Amazon CloudWatch Metrics](#).

- Pour les procédures de journalisation du débogage, consultez [Génération de journaux de débogage](#).

Résolution des problèmes de lecture depuis MediaTailor

Cette section fournit des informations sur les codes d'erreur HTTP que vous pouvez recevoir lorsque vous testez le logiciel de votre lecteur au cours du traitement normal des demandes du lecteur.

Note

Vous pouvez également recevoir des erreurs de la part de l' AWS Elemental MediaTailor API, lors d'opérations de configuration telles que `PutPlaybackConfiguration` et `GetPlaybackConfiguration`. Pour plus d'informations sur ces types d'erreurs, consultez la référence de l'[AWS Elemental MediaTailor API](#).

Lorsque votre joueur envoie une demande à AWS Elemental MediaTailor, directement ou via un CDN, il MediaTailor répond par un code d'état. En cas de gestion MediaTailor réussie de la demande, elle renvoie le code d'état HTTP200 OK, indiquant le succès, ainsi que le manifeste rempli. Si la demande échoue, MediaTailor renvoie un code d'état HTTP, un nom d'exception et un message d'erreur.

AWS Elemental MediaTailor renvoie deux classes d'erreurs :

- Erreurs du client : erreurs qui sont généralement causées par un problème lié à la demande elle-même, comme une demande mal formatée, un paramètre non valide ou une URL incorrecte. Ces erreurs ont un code de réponse 4xx HTTP.
- Erreurs de serveur : erreurs généralement causées par un problème lié à ses dépendances MediaTailor ou à l'une de ses dépendances, comme le serveur de décision publicitaire (ADS) ou le serveur d'origine. Ces erreurs ont un code de réponse 5xx HTTP.

Rubriques

- [Erreurs de lecture du client renvoyées par AWS Elemental MediaTailor](#)
- [Erreurs de lecture sur le serveur renvoyées par AWS Elemental MediaTailor](#)
- [Exemples d'erreurs de lecture](#)

Erreurs de lecture du client renvoyées par AWS Elemental MediaTailor

Directives générales :

- Vous trouverez des informations détaillées sur la plupart des erreurs dans les en-têtes et le corps de la réponse.
- Pour détecter certaines erreurs, vous devez vérifier vos paramètres de configuration. Vous pouvez récupérer les paramètres de votre configuration de lecture à l'adresse AWS Elemental MediaTailor. Pour l'API, la ressource est `GetPlaybackConfiguration/Name`. Pour plus de détails, consultez la référence de [AWS Elemental MediaTailor l'API](#).

Le tableau suivant répertorie les codes d'erreur client renvoyés par les activités de manipulation du manifeste AWS Elemental MediaTailor, les causes probables et les mesures que vous pouvez prendre pour les résoudre.

Code	Nom de l'exception	Signification	Que faire
400	<code>BadRequestException</code>	MediaTailor n'est pas en mesure de traiter la demande en raison d'une ou de plusieurs erreurs de mise en forme ou de contenu. Un paramètre est peut-être mal formaté ou la demande peut contenir une configuration de lecture ou un ID de session non valide.	Vérifiez que votre demande est correctement formatée et contient des informations précises. Assurez-vous que le paramètre du point de terminaison de lecture sur le lecteur correspond au paramètre <code>ManifestEndpointPrefix</code> renvoyé par <code>GetPlaybackConfiguration</code> . Réessayez votre demande.
400	<code>AccessDeniedException</code>	L'en-tête d'hôte fourni dans la demande ne correspond pas au préfixe de point de terminaison de manifeste configuré dans l'URL de lecture de MediaTailor. Votre CDN est peut-être mal configuré.	Vérifiez les paramètres du CDN et assurez-vous que vous utilisez le préfixe de point de terminaison de manifeste correct pour MediaTailor. Réessayez votre demande.

Code	Nom de l'exception	Signification	Que faire
40	NotFoundException	MediaTailor est incapable de trouver les informations spécifiées. Les raisons possibles incluent une URL qui ne mappe pas à quoi que ce soit dans le service, une configuration qui n'est pas définie, ou une session qui n'est pas disponible.	Vérifiez votre configuration et la validité de votre demande, puis réinitialisez la session.
40	ConflictException	Un lecteur a essayé de charger plusieurs listes de lecture simultanément pour une seule session. Par conséquent, un conflit de cohérence de session a MediaTailor été détecté. Ce problème se produit pour les lecteurs HLS.	Assurez-vous que votre lecteur demande les listes de lecture une à la fois. C'est conforme à la spécification HLS.
41	Gone	Un opérateur AWS du support a bloqué une session de joueur ou une configuration client. AWS Le support le fait dans de rares circonstances lorsque nous détectons un très grand volume de requêtes 4xx provenant d'un trafic errant pour une seule session ou configuration.	Si vous pensez que la demande ne doit pas être bloquée, contactez AWS Support . Ils peuvent vérifier et supprimer le filtre à l'origine du blocage, le cas échéant.

Si vous avez besoin d'une assistance supplémentaire, contactez [AWS Support](#).

Erreurs de lecture sur le serveur renvoyées par AWS Elemental MediaTailor

Directives générales :

- Vous trouverez des informations détaillées sur la plupart des erreurs dans les en-têtes et le corps de la réponse.
- Pour détecter certaines erreurs, vous devez vérifier vos paramètres de configuration. Vous pouvez récupérer les paramètres de votre configuration de lecture à l'adresse AWS Elemental MediaTailor. Pour l'API, la ressource est `GetPlaybackConfiguration/Name`. Pour plus de détails, consultez la référence de [AWS Elemental MediaTailor l'API](#).

Le tableau suivant répertorie les codes d'erreur du serveur renvoyés par les activités de manipulation du manifeste AWS Elemental MediaTailor, les causes probables et les mesures que vous pouvez prendre pour les résoudre.

Code	Nom de l'exception	Signification	Que faire
50	InternalServerError	Exception non gérée	Réitérez la demande. Si le problème persiste, vérifiez l'état de santé déclaré MediaTailor pour votre AWS région à l'adresse https://status.aws.amazon.com/ .
50	BadGatewayException	L'adresse du serveur d'origine ou celle du serveur de décision publicitaire (ADS) n'est pas valide. Des exemples d'adresses non valides sont une adresse IP privée et localhost .	Assurez-vous que votre configuration possède les paramètres appropriés pour votre ADS et le serveur d'origine, puis réessayez la demande.
50	UnsupportedManifestException	Soit le manifeste d'origine a changé de sorte qu'il ne MediaTailor peut pas être personnalisé, soit il MediaTailor ne prend pas en charge le format du manifeste d'origine.	Cela peut affecter uniquement une session individuelle. Réinitialisez la session. Vous pouvez généralement effectuer cette opération en actualisant la page dans le lecteur. Si le problème persiste, vérifiez que MediaTailor prend en charge le format du manifeste d'origine. Pour

Code	Nom de l'exception	Signification	Que faire
			plus d'informations, consultez Intégrer une source de contenu.
50	LoadShed	MediaTailor vous avez rencontré une contrainte de ressources lors du traitement de votre demande.	Réitérez la demande. Si le problème persiste, vérifiez l'état de santé déclaré MediaTailor pour votre AWS région à l'adresse https://status.aws.amazon.com/.
50	ThrottlingException	Vos transactions par seconde ont atteint votre quota, et MediaTailor limite votre utilisation.	Réitérez la demande. Vous pouvez également consulter l'état de santé déclaré MediaTailor pour votre AWS région à l'adresse https://status.aws.amazon.com/. Vous pouvez augmenter le quota sur vos transactions par seconde. Pour de plus amples informations, veuillez consulter the section called "Quotas relatifs à l'insertion d'annonces".
50	GatewayTimeoutException	Un délai d'attente s'est produit alors que MediaTailor je contactais le serveur d'origine.	Réitérez la demande. Si le problème persiste, vérifiez l'état du serveur d'origine et assurez-vous que le serveur d'origine répond dans le délai d'attente du serveur d'origine du contenu répertorié à l'adresse the section called "Quotas relatifs à l'insertion d'annonces".

Si vous avez besoin d'une assistance supplémentaire, contactez [AWS Support](#).

Exemples d'erreurs de lecture

Cette section répertorie quelques exemples d'erreurs de lecture que vous pouvez rencontrer lors des interactions en ligne de commande avec AWS Elemental MediaTailor.

L'exemple suivant montre le résultat lorsqu'un délai d'attente se produit entre AWS Elemental MediaTailor le serveur de décision publicitaire (ADS) ou le serveur d'origine.

```
~[ ]> curl -vvv https://111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com/v1/master/123456789012/Multiperiod_DASH_Demo/index.mpd
* Trying 54.186.133.224...
* Connected to 111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com (11.222.333.444) port 555 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
* Server certificate: mediataylor.us-west-2.amazonaws.com
* Server certificate: Amazon
* Server certificate: Amazon Root CA 1
* Server certificate: Starfield Services Root Certificate Authority - G2
> GET /v1/master/123456789012/Multiperiod_DASH_Demo/index.mpd HTTP/1.1
> Host: 111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 504 Gateway Timeout
< Date: Thu, 29 Nov 2018 18:43:14 GMT
< Content-Type: application/json
< Content-Length: 338
< Connection: keep-alive
< x-amzn-RequestId: 123456789012-123456789012
< x-amzn-ErrorType: GatewayTimeoutException:http://internal.amazon.com/coral/com.amazon.elemental.midas.mms.coral/
<
* Connection #0 to host 111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com left intact
{"message":"failed to generate manifest: Unable to obtain template playlist. origin URL:[https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/444455556666111122223333/index.mpd], asset path: [index.mpd], sessionId:[123456789012123456789012] customerId:[123456789012]"}%
```

Déduplication des balises

Cette section fournit des conseils de dépannage pour MediaTailor la fonction de déduplication des balises dans le suivi côté serveur.

Symptôme : nombre de balises inférieur à celui observé précédemment

Cause : La déduplication des balises empêche les événements dupliqués qui étaient précédemment comptabilisés plusieurs fois.

Résolution : ce comportement est attendu et indique une mesure plus précise. Les nouveaux chiffres représentent les événements réels de visionnage des publicités plutôt que les demandes de segments techniques. Mettez à jour vos données de référence pour refléter les chiffres dédupliés.

Symptôme : le nombre de balises varie selon les implémentations des joueurs

Cause : Les modèles de demandes de segments peuvent varier d'un acteur à l'autre, mais la déduplication garantit la cohérence du nombre de balises, quelles que soient ces différences.

Résolution : il s'agit d'un comportement attendu. La déduplication fournit des mesures cohérentes pour les différents types de joueurs et les conditions du réseau.

Symptôme : écart entre le nombre de balises attendu et réel

Cause : Les prévisions précédentes étaient peut-être fondées sur des chiffres gonflés en raison du double déclenchement de balises.

Solution : Passez en revue les données historiques pour identifier les modèles de balises dupliquées. Utilisez les nouveaux chiffres dédupliés comme base de référence précise pour les mesures et les rapports.

Résolution des problèmes de paramètres

Pour résoudre les problèmes liés aux variables publicitaires dynamiques, aux paramètres de requête du manifeste, aux restrictions de caractères, aux limites de longueur et aux alias de configuration, voir [MediaTailor guide de dépannage des paramètres](#) dans [MediaTailor variables publicitaires dynamiques pour les demandes ADS](#).

MediaTailor guide de résolution des problèmes liés à l'omission d'annonces

Le fait de sauter une annonce est l'un des problèmes les plus courants signalés par les MediaTailor clients. Ce guide vous aide à identifier les problèmes liés au fait d'ignorer les publicités et propose des procédures de dépannage étape par étape.

Symptômes et impact

Lorsque vous ignorez une annonce, vous pouvez observer les symptômes suivants :

- Les publicités ne s'affichent pas pendant les pauses publicitaires prévues
- Les pauses publicitaires diffusent du contenu au lieu de publicités
- Diffusion incohérente des publicités entre les différentes sessions de visionnage
- CloudWatch journaux affichant les AdSkipped événements pour lesquels diverses raisons peuvent être ignorées

Le fait de sauter des publicités a un impact direct sur les revenus et peut nuire à l'expérience des spectateurs. Résolvez rapidement ces problèmes pour maintenir des performances optimales.

Identifier les raisons des sauts

Utilisez CloudWatch Logs Insights pour interroger le groupe de MediaTailor/AdDecisionServerInteractions journaux pour les annonces ignorées :

```
fields @timestamp, avail.availId, skippedAds.0.skippedReason,
skippedAds.0.creativeUniqueId
| filter eventType = "FILLED_AVAIL" and ispresent(skippedAds.0.skippedReason)
| sort @timestamp desc
```

Cette requête renvoie les événements d'interruption des publicités les plus récents, avec leurs raisons spécifiques, afin de vous aider à identifier les tendances.

Rubriques

- [MediaTailor Résolution des problèmes liés à l'utilisation de NEW_CREATIVE](#)
- [MediaTailor Résolution des problèmes liés au délai d'attente et aux sauts d'ADS](#)

- [MediaTailor Résolution des problèmes d'analyse et de sauts VAST](#)
- [MediaTailor résolution des problèmes d'inadéquation de durée et de sauts](#)
- [MediaTailor résolution des problèmes liés aux variables de session et aux sauts](#)
- [MediaTailor surveillance et alertes en cas d'oubli des publicités](#)
- [MediaTailor les meilleures pratiques en matière de prévention des sauts publicitaires](#)
- [MediaTailor guide de référence pour ignorer les publicités](#)

MediaTailor Résolution des problèmes liés à l'utilisation de NEW_CREATIVE

Lorsque des publicités sont ignorées avec la NEW_CREATIVE raison, une annonce nécessitant AWS Elemental MediaTailor un transcodage est détectée avant d'être insérée. Ce guide de dépannage explique les causes et fournit des procédures de résolution étape par étape.

Quelles sont les causes du saut de NEW_CREATIVE

MediaTailor transcode les publicités en fonction de trois facteurs clés : l'identifiant de création, l'identifiant de compte AWS et l'ensemble de variantes de transcodage (les rendus de lecture pour le flux de contenu sous-jacent). Lorsqu'une partie de l'identifiant créatif ou de l'ensemble de variantes de transcodage diffère, MediaTailor reconnaît l'annonce comme une nouvelle variante nécessitant un transcodage.

Changements de débit

Les changements de débit peuvent entraîner le saut de NEW_CREATIVE lorsque le débit ne correspond pas après avoir été arrondi aux 8 000 bits les plus proches. Cela se produit généralement dans les situations suivantes :

- Le flux de contenu sous-jacent modifie le manifeste principal
- Les nouvelles sessions sont créées avec des variantes différentes des sessions existantes
- Les débits des flux de contenu ne sont pas cohérents entre les sessions de lecture

Variance de bande passante des tâches de récolte

Pour les tâches de récolte, MediaTailor utilise un seuil de variation de bande passante de 15 % lors de la mise en correspondance des publicités avec des flux de contenu. S'il existe une différence de 15 % ou plus entre la bande passante de la diffusion en direct et la bande passante du fichier

HLS récolté, l'annonce sera à nouveau transcodée. Cela peut entraîner le saut de NEW_CREATIVE pendant le processus de transcodage.

Ce comportement se produit généralement lorsque :

- Les mêmes publicités pré-transcodées sont utilisées à la fois pour la diffusion en direct et le contenu VOD
- Le contenu récolté possède des paramètres de codage différents de ceux du flux en direct d'origine
- Les différences de bande passante dépassent le seuil de 15 % entre le contenu en direct et le contenu récolté

Conflits entre Creative ID

Lorsqu' MediaTailor un identifiant créatif différent est détecté pour un fichier multimédia déjà transcodé, la séquence suivante se produit :

1. L'annonce est ignorée pour la raison NEW_CREATIVE
2. Cela entraîne une tentative de transcodage inutile
3. La création est marquée comme DUPLICATE_TRANSCODE ou COPY_DEDUP

Note

MediaTailor n'expire pas et ne supprime pas les publicités transcodées. Nous les stockons indéfiniment dans un compartiment MediaTailor-owned S3.

Étapes de résolution

Pour résoudre les problèmes liés à l'omission d'annonces dans NEW_CREATIVE, procédez comme suit :

1. Vérifiez que votre serveur de décision publicitaire renvoie des identifiants créatifs cohérents pour le même contenu publicitaire.
2. Vérifiez que votre flux de contenu conserve des débits et des ensembles de variantes cohérents.
3. Envisagez de mettre en œuvre la prélecture des publicités pour vous assurer que les publicités sont transcodées avant leur diffusion. Pour de plus amples informations, veuillez consulter [the section called “Publicités de prélecture”](#).

4. En cas de problèmes persistants, contactez [AWS Support](#) pour obtenir une assistance de dépannage supplémentaire.

Surveillance des modèles NEW_CREATIVE

Utilisez cette requête CloudWatch Logs Insights pour analyser les tendances dans NEW_CREATIVE en matière de saut d'annonces :

```
fields @timestamp, sessionId, creativeId, skipReason, MediaFileSourceUrl
| filter skipReason = "NEW_CREATIVE"
| stats count() by creativeId, MediaFileSourceUrl
| sort count desc
| limit 50
```

MediaTailor Résolution des problèmes liés au délai d'attente et aux sauts d'ADS

Lorsque des publicités sont ignorées ADS_TIMEOUT pour des raisons connexes, vous rencontrez des problèmes de connectivité ou de performances avec votre serveur de décision publicitaire. AWS Elemental MediaTailor nécessite une communication fiable avec votre ADS pour réussir à insérer des annonces. Ce guide de dépannage explique comment identifier et résoudre ces problèmes de connectivité.

Problèmes courants de connectivité ADS

Les problèmes de connectivité ADS courants sont les suivants :

- Le serveur de décision publicitaire n'est pas accessible depuis MediaTailor
- ADS ne répond pas dans le délai d'attente configuré
- ADS n'est pas en mesure de gérer le volume de demandes pendant les périodes de pointe
- Problèmes de connectivité réseau entre MediaTailor et votre ADS

Étapes de résolution

Pour résoudre les problèmes de délai d'expiration de l'ADS :

1. Vérifiez que votre serveur de décision publicitaire est accessible depuis MediaTailor.
2. Vérifiez que votre ADS répond dans le délai d'attente configuré.
3. Assurez-vous que votre ADS peut gérer le volume de demandes pendant les périodes de pointe.
4. Envisagez de mettre en œuvre une stratégie publicitaire de repli lorsque votre ADS principal n'est pas disponible.

Surveillance des performances de l'ADS

Configurez CloudWatch des alarmes pour la `AdDecisionServer.Timeouts` métrique afin de détecter de manière proactive les problèmes de connectivité ADS.

MediaTailor Résolution des problèmes d'analyse et de sauts VAST

Lorsque les publicités sont ignorées `VAST_PARSING_ERROR` ou `MEDIA_FILE_UNAVAILABLE` que vous rencontrez des problèmes avec vos réponses VAST ou vos fichiers multimédia publicitaires. AWS Elemental MediaTailor nécessite des réponses VAST correctement formatées et des fichiers multimédia accessibles pour une insertion d'annonce réussie. Ce guide de dépannage explique comment identifier et résoudre ces problèmes.

Problèmes VAST courants

Les problèmes VAST courants sont les suivants :

- Le format de réponse VAST n'est pas conforme à la spécification VAST
- Les URL des fichiers multimédia dans la réponse VAST ne sont pas accessibles au public
- Les caractères spéciaux ne sont pas correctement encodés dans VAST XML
- Réponse VAST manquante pour les fichiers ou formats multimédia requis

Étapes de résolution

Pour résoudre les problèmes d'analyse de VAST :

1. Validez votre format de réponse VAST par rapport à la spécification VAST.
2. Assurez-vous que toutes les URL des fichiers multimédia de la réponse VAST sont accessibles au public.
3. Vérifiez que le codage des caractères spéciaux est correct dans votre fichier VAST XML.

4. Vérifiez que votre réponse VAST inclut des fichiers multimédia dans des formats compatibles avec MediaTailor.

Résolution des problèmes liés au wrapper VAST

En cas de problème avec `INVALID_VAST_WRAPPER_AD` ou `REJECTED_REPLICA_VAST` :

- Valider les réponses du wrapper VAST par rapport à la spécification VAST
- Assurez-vous que tous les éléments du wrapper sont correctement formatés et contiennent des éléments valides `VASTAdTagURI`
- Vérifiez la configuration du serveur publicitaire pour connaître les politiques de détection du contenu dupliqué
- Assurez-vous que les réponses VAST contiennent un contenu créatif unique au cours de la même session

MediaTailor résolution des problèmes d'inadéquation de durée et de sauts

Lorsque des publicités sont ignorées ou pour des raisons liées au format `AVAIL_DURATION_EXCEEDEDLEFTOVER_AVAIL_EXCEEDED_THRESHOLD`, vous rencontrez des problèmes de durée ou de compatibilité des formats. AWS Elemental MediaTailor nécessite un formatage de la durée et une correspondance appropriés entre le contenu de l'annonce et la durée de pause disponible. Ce guide de dépannage explique comment identifier et résoudre ces problèmes liés à la durée.

Problèmes de durée courants

Les problèmes de durée courants sont les suivants :

- Publicités dont la durée d'interruption est supérieure à celle disponible
- Marqueurs d'interruption de publicité incorrects dans le contenu signalant une durée incorrecte
- Seuil de personnalisation non atteint pour l'insertion d'annonces
- Format de durée incorrect dans les `EXT-X-CUE-OUT` balises

Exigences relatives au format de durée

Le paramètre de durée de la EXT-X-CUE-OUT balise doit être formaté comme une valeur entière, et non comme un format de durée ISO 8601.

Exigences relatives au format de durée

Format	Exemple	Statut
Entier (correct)	32	Pris en charge : représente 32 secondes
Décimal (correct)	30.000	Pris en charge : représente 30 secondes
ISO 8601 (incorrect)	PT32S	Non pris en charge : entraîne des échecs d'insertion

Étapes de résolution

Pour résoudre les problèmes d'inadéquation de durée :

1. Assurez-vous que votre ADS renvoie des annonces correspondant à la durée de pause publicitaire disponible.
2. Vérifiez que vos marqueurs d'interruption de publicité dans le contenu indiquent correctement la durée prévue.
3. Vérifiez que les paramètres de EXT-X-CUE-OUT durée utilisent le format entier.
4. Envisagez d'ajuster le seuil de personnalisation en fonction de votre cas d'utilisation.

MediaTailor résolution des problèmes liés aux variables de session et aux sauts

Les variables de session jouent un rôle essentiel dans le ciblage et la sélection des publicités dans AWS Elemental MediaTailor. Une configuration incorrecte des variables de session est une cause fréquente de problèmes liés à l'oubli des publicités. Ce guide de dépannage complet explique comment identifier et résoudre les problèmes liés aux variables de session susceptibles d'empêcher l'insertion d'annonces.

Problèmes courants liés aux variables de session

Les problèmes courants liés aux variables de session sont les suivants :

- Variables obligatoires manquantes : votre serveur de décision publicitaire peut avoir besoin de variables spécifiques qui ne sont pas fournies.
- Syntaxe de variable incorrecte : les variables doivent utiliser la syntaxe correcte (par exemple, `[session.id]` au lieu de `${session.id}`).
- Problèmes de codage d'URL : les caractères spéciaux contenus dans les valeurs des variables peuvent nécessiter un encodage d'URL approprié.
- Paramètres de joueur incohérents : les paramètres du joueur doivent être transmis de manière cohérente d'une session à l'autre.
- Défaillances de résolution des variables dynamiques : les variables qui ne peuvent pas être résolues sont remplacées par des chaînes vides.
- SCTE-35 Problèmes liés à l'analyse des données UPID : les problèmes liés au traitement des données UPID par segmentation peuvent entraîner des échecs de résolution des variables de session.

Vérification de la résolution des variables de session

Pour vérifier que vos variables de session sont correctement résolues :

1. Activez la journalisation du débogage pour votre configuration MediaTailor
2. Vérifiez le groupe de MediaTailor/AdDecisionServerInteractions journaux pour les URL réelles des requêtes ADS
3. Vérifiez que toutes les variables de l'URL du modèle ont été remplacées par des valeurs appropriées
4. Recherchez toutes les variables qui ont été remplacées par des chaînes vides, ce qui peut indiquer des échecs de résolution

SCTE-35 Résolution des problèmes liés à l'analyse des UPID

Les problèmes liés au traitement UPID de SCTE-35 segmentation peuvent entraîner des problèmes liés aux variables de session :

- Exigences de format : l'UPID doit avoir une valeur `segmentation_upid_type` de 12 et être inclus `format_identifieur` pour un traitement correct
- Règles d'analyse : l'UPID décodé peut contenir des séparateurs deux-points pour plusieurs valeurs. Le nombre de variables de modèle et de jetons UPID décodés doit être égal
- Formats non valides : évitez les doubles virgules sans valeur (par exemple, `::` ou `:46175218::4053`) car ils entraînent des échecs d'analyse

MediaTailor surveillance et alertes en cas d'oubli des publicités

La surveillance proactive vous permet de détecter et de résoudre les problèmes d'omission des publicités avant qu'ils n'aient un impact significatif sur vos revenus. AWS Elemental MediaTailor fournit des mesures complètes et des fonctionnalités de journalisation qui permettent de surveiller efficacement les performances d'insertion d'annonces. Ce guide explique comment mettre en place une surveillance efficace des problèmes liés au fait d'ignorer les publicités.

CloudWatch Indicateurs clés à surveiller

Configurez CloudWatch des alarmes pour les MediaTailor indicateurs clés suivants :

- `AdDecisionServer.Ads.Skipped`- Nombre d'annonces ignorées
- `AdDecisionServer.Timeouts`- Nombre de délais d'attente de l'ADS
- `Avail.FilledDuration`- Durée des pauses publicitaires
- `Avail.SlateOnly`- Nombre de pauses publicitaires remplies uniquement avec de l'ardoise

Requêtes Advanced CloudWatch Logs Insights

Utilisez ces requêtes spécialisées pour une résolution détaillée des problèmes :

Analyse complète des sessions

Pour une analyse détaillée du comportement d'insertion d'annonces pour une session spécifique, procédez comme suit :

```
fields @timestamp, sessionId, eventType, creativeId, skipReason, adBreakIndex
| filter sessionId = "your-session-id-here"
| filter eventType in ["FILLED_AVAIL", "SKIPPED_AVAIL", "MAKING_ADS_REQUEST"]
```

```
| sort @timestamp asc  
| limit 100
```

Trouver des Creative ID

Pour identifier les Creative ID à partir des événements FILLED_AVAIL, procédez comme suit :

```
fields @timestamp, sessionId, eventType  
| filter sessionId like /sessionId/ and eventType!='BEACON_FIRED'  
| sort @timestamp desc
```

Note

sessionIdRemplacez-le par l'identifiant de session réel que vous recherchez.

MediaTailor les meilleures pratiques en matière de prévention des sauts publicitaires

La mise en œuvre de ces meilleures pratiques permet d'éviter les problèmes d'omission des publicités avant qu'ils ne surviennent, garantissant ainsi de meilleures performances d'insertion d'annonces et une AWS Elemental MediaTailor protection des revenus. Ces mesures proactives s'attaquent aux causes les plus courantes de sauts d'annonces et contribuent à maintenir une diffusion cohérente des publicités.

Mesures proactives

- Implémentez la prélecture des publicités : utilisez MediaTailor la fonction de prélecture des publicités pour vous assurer que les publicités sont transcodées avant leur diffusion. Voir les détails [the section called “Publicités de prélecture”](#) de mise en œuvre
- Maintenez des identifiants créatifs cohérents : assurez-vous que votre serveur de décision publicitaire utilise des identifiants créatifs cohérents pour le même contenu publicitaire d'une session à l'autre
- Assurez-vous que le formatage de la durée est correct : utilisez des valeurs entières pour les paramètres de EXT-X-CUE-OUT durée au lieu du format ISO 8601

- Configurer l'optimisation de la VOD : définissez le paramètre Maximum de requêtes ADS simultanées pour les flux VOD comportant plusieurs pauses publicitaires afin de réduire la charge du serveur. Pour de plus amples informations, consultez [Réglages avancés](#).
- Optimisez les performances ADS : configurez votre serveur de décision publicitaire pour qu'il réponde rapidement et gère les pics de trafic

Directives d'implémentation

- Mettez en œuvre une gestion des erreurs appropriée : configurez le contenu de l'ardoise pour qu'il remplisse les pauses publicitaires lorsque les publicités ne peuvent pas être insérées
- Testez minutieusement : validez votre flux de travail d'insertion d'annonces sur différents appareils et conditions de réseau
- Mettez en œuvre des stratégies de secours : configurez des sources publicitaires de sauvegarde ou des publicités par défaut en cas d'échec des sources publicitaires principales
- Surveillez les modèles de transcodage : surveillez CloudWatch les journaux pour vérifier l'efficacité du transcodage et contactez [AWS Support](#) si vous remarquez des modèles indiquant des problèmes de transcodage

MediaTailor guide de référence pour ignorer les publicités

Cette section fournit des informations de référence complètes sur les raisons de l'exclusion des publicités et des liens vers la documentation associée à AWS Elemental MediaTailor. Utilisez ce guide de référence pour comprendre la signification spécifique des différentes raisons de sauter et trouver des ressources de dépannage supplémentaires.

Référence complète des motifs d'exclusion des publicités

MediaTailor enregistre les raisons spécifiques pour lesquelles les publicités sont ignorées dans le message du journal des FILLED_AVAIL événements du groupe de MediaTailor/AdDecisionServerInteractions journaux.

Raisons complètes pour ignorer les publicités

Ignorer la raison	Description
NEW_CREATIVE	L'annonce n'a pas encore été transcodée. Cela se produit lorsque MediaTailor vous rencontrez une nouvelle création publicitaire qui nécessite un transcodage avant son insertion.
PROFILE_NOT_FOUND	Le profil de MediaConvert transcodage associé à la configuration de la session n'existe pas, ce qui empêche la préparation des publicités.
TRANSCODE_ERROR	Le processus de transcodage publicitaire a rencontré une erreur et n'a pas pu s'achever.
TRANSCODE_IN_PROGRESS	Le transcode publicitaire est toujours en cours et n'est pas encore prêt à être inséré.
INTERNAL_ERROR	Une MediaTailor erreur interne s'est produite lors de la gestion de l'annonce, empêchant son insertion.
AVAIL_DURATION_EXCEEDED	L'annonce ne correspond pas à la durée restante de la pause publicitaire.
LEFTOVER_AVAIL_EXCEEDED_THRESHOLD	La durée cumulée de toutes les publicités qui auraient pu être insérées n'atteint pas le seuil de personnalisation configuré pour la session.
VAST_PARSING_ERROR	La réponse VAST du serveur de décision publicitaire contient des erreurs ou est mal formée.
ADS_TIMEOUT	Le serveur de décision publicitaire n'a pas répondu dans le délai d'expiration configuré.
MEDIA_FILE_UNAVAILABLE	Les fichiers publicitaires spécifiés dans la réponse VAST ne sont pas accessibles.
SESSION_INITIALIZATION_FAILED	La MediaTailor session n'a pas pu s'initialiser correctement, souvent en raison de variables de session incorrectes.

Ignorer la raison	Description
EARLY_CUE_IN	La pause publicitaire s'est terminée plus tôt que prévu en raison d'un signal de repérage précoce, empêchant l'insertion complète de l'annonce.
NO_VARIANT_MATCH	La création publicitaire ne possède pas de variante correspondant aux paramètres de codage du flux de contenu (débit, résolution, codec).
NO_MODEL_CREATIVE_MATCH	La création publicitaire ne correspond pas aux exigences de modèle ou de format attendues pour la configuration de lecture actuelle.
REJECTED_REPLICA_VAST	La réponse VAST a été rejetée en raison de politiques de détection de répliques ou de doublons de contenu.
INVALID_VAST_WRAPPER_AD	L'annonce d'emballage VAST contient des éléments d'emballage non valides ou mal formés qui empêchent l'insertion réussie de l'annonce.
IMPORT_ERROR	Une erreur s'est produite lors du processus d'importation de l'annonce, empêchant le traitement de l'annonce en vue de son insertion.
IMPORT_IN_PROGRESS	Le processus d'importation des publicités est en cours et n'est pas encore terminé.

Ressources connexes

Pour plus d'informations sur la résolution des problèmes liés au fait d'ignorer les publicités, consultez les rubriques connexes suivantes :

- [the section called “Variables publicitaires dynamiques”](#) - Guide complet des variables publicitaires dynamiques dans MediaTailor
- [the section called “Publicités de prélecture”](#) - Comment implémenter la prélecture des annonces pour éviter les sauts liés au transcodage
- [the section called “Surveillance à l'aide de CloudWatch métriques”](#) - Surveillance à l'aide de CloudWatch métriques
- [the section called “Affichage des journaux”](#) - Comment consulter et analyser les journaux MediaTailor

- [the section called “Résolution des problèmes de flux d'événements”](#) - Comprendre le flux des événements d'insertion d'annonces

AWS Elemental MediaTailor ressources

Le tableau suivant répertorie les ressources connexes qui pourront vous être utiles lors de l'utilisation de AWS Elemental MediaTailor.

Ressource	Description
Norme SCTE : SCTE 35	Le document standard de la SCTE pour SCTE35.
Cours et ateliers	Des liens vers des cours spécialisés et basés sur les rôles ainsi que des ateliers d'autoformation pour vous aider à perfectionner vos AWS compétences et à acquérir une expérience pratique.
AWS outils de développement	Liens vers des outils de développement SDKs, des kits d'outils IDE et des outils de ligne de commande pour le développement et la gestion AWS d'applications.
AWS livres blancs	Liens vers une liste complète de AWS livres blancs techniques, traitant de sujets tels que l'architecture, la sécurité et l'économie et rédigés par des architectes de AWS solutions ou d'autres experts techniques.
AWS Centre de support	La plateforme de création et de gestion de vos dossiers de AWS Support. Comprend également des liens vers d'autres ressources utiles, telles que des forums, des informations techniques FAQs, l'état de santé des services et AWS Trusted Advisor.
AWS Support	La page Web principale contenant des informations sur le AWS Support one-on-one, un canal

Ressource	Description
	d'assistance rapide destiné à vous aider à créer et à exécuter des applications dans le cloud.
Contactez-nous	Point de contact central pour les demandes concernant la AWS facturation, le compte, les événements, les abus et autres problèmes.
AWS termes du site	Informations détaillées sur nos copyrights et marque de commerce, sur votre compte, votre licence et votre accès au site et d'autres sujets.

Historique des documents pour AWS Elemental MediaTailor

Le tableau suivant décrit les modifications importantes apportées à la documentation.

Modification	Description	Date
Ajout du délai d'expiration ADS, du budget temporel de personnalisation et des paramètres de simultanéité	Ajout d'une documentation sur les délais d'expiration des requêtes configurables du serveur de décision publicitaire (ADS), les budgets temporels de personnalisation et la simultanéité des configurations de lecture, y compris les nouveaux paramètres de délai de prélecture.	23 juillet 2026
Account-level dimensions pour des raisons d'annonce ignorée	Ajout d'émissions au niveau du compte (sans dimensions) pour les SkippedReason métriques.	21 juillet 2026
Nouvelles dimensions métriques et au niveau du compte	La Avail.Complete métrique a été ajoutée. Ajout d'émissions au niveau du compte (sans dimensions) pour lesRequests,AdsBilled,Avail.Impression,Avail.CompleteAvail.FilledDurationAvail.Duration,etAdDecisionServer.Ads les métriques.	30 juin 2026
Authentification automatique côté serveur avec les	Le chapitre a été mis à jour pour refléter le fait que les	15 avril 2026

[plateformes publicitaires de Google](#)

connexions de serveur à serveur authentifient MediaTailor désormais automatiquement les connexions avec Google Ad Manager (GAM), Google Campaign Manager (GCM) et Google Display & Video 360 (DV360). Aucune action du client ou ticket d'assistance n'est requis. Mise à jour de la section côté serveur pour décrire les URI-based règles MediaTailor applicables et suppression de la procédure manuelle de ticket d'assistance.

[Ajout de la mise en forme du trafic TPS](#)

Sections mises à jour pour inclure des informations sur la limitation des demandes à l'ADS en fonction du TPS.

2 octobre 2025

[Ajouté prefetchScheduleName](#)

Ajout d'informations sur prefetchScheduleName les journaux ADS.

24 septembre 2025

[Guide de résolution des problèmes liés à la restructuration des publicités](#)

Guide de dépannage restructuré pour concentrer chaque section sur une tâche spécifique à effectuer.

26 août 2025

Ajout de la documentation sur la gestion des paramètres de requête	Ajout d'une section expliquant comment MediaTailor gère les paramètres de requête pour les fonctionnalités spécifiques à l'origine, comme l'affichage décalé dans le temps avec MediaPackage	26 août 2025
Guide de configuration SGAI VOD	Ajout de conseils de configuration pour l'insertion d'annonces guidée par le serveur avec du contenu vidéo à la demande.	7 août 2025
Guide intégré de dépannage des paramètres	Ajout d'une section de dépannage complète couvrant les erreurs de restriction de caractères, les problèmes de limitation de longueur, les problèmes d'alias de configuration, le débogage du flux de traitement des paramètres et les meilleures pratiques de sécurité.	4 août 2025
Documentation améliorée sur le formatage des paramètres	Ajout de restrictions de caractères complètes, de URL-encoding prise en charge et de limitations de longueur pour les paramètres de requête du manifeste et les paramètres ADS.	1er août 2025

Documentation des paramètres améliorée avec référence complète	Ajout de restrictions complètes concernant les caractères des paramètres, de limites de longueur, d'une URL-encoding assistance, d'une documentation sur les alias de configuration et de méthodes d'initialisation de session basées sur l'analyse d'AWS Guide.	1er août 2025
Documentation améliorée sur les paramètres de requête du manifeste	Ajout d'un comportement complet de placement des paramètres, de restrictions de caractères, de URL-encoding prise en charge et de différences spécifiques au protocole sur la base de l'analyse d'AWS Guide.	1er août 2025
Documentation améliorée sur les variables de domaine	Ajout d'une documentation complète sur les alias de configuration, de champs pris en charge et de règles de paramètres au niveau du domaine basées sur l'analyse d'AWS Guide.	1er août 2025
Ajout d'un guide de dépannage du flux d'événements	Nouvelle section fournissant des conseils pratiques sur l'utilisation de l'analyse des flux d' MediaTailor événements pour diagnostiquer et résoudre les problèmes d'insertion d'annonces.	31 juillet 2025

Ajout de la documentation sur le flux d'événements d'insertion d'annonces	Nouvelle section expliquant le flux chronologique des MediaTailor événements lors de la personnalisation du manifeste afin d'aider les utilisateurs à comprendre le flux de travail.	23 juillet 2025
Documentation améliorée sur le flux de travail de suivi côté client	Ajout d'une documentation complète sur le flux de travail de reporting côté client avec un diagramme de séquence détaillé et une explication étape par étape du processus.	16 juillet 2025
Documentation sur l'intégration du CDN	Ajout d'informations plus complètes sur la configuration d'un CDN avec MediaTailor.	30 juin 2025
Ajout de l'intégration CDN pour Channel Assembly	Ajout d'une documentation pour intégrer un CDN à Channel Assembly afin d'optimiser la diffusion du contenu.	16 mai 2025
Types de listes de lecture HLS	Ajout de définitions pour les listes de lecture multivariantes et multimédia HLS.	9 mai 2025
Types de manifestes DASH	Ajout de définitions pour les composants du manifeste DASH.	9 mai 2025
Ajout d'informations sur le type de playlist HLS	Ajout d'explications sur les listes de lecture multivariantes et variantes pour fournir un contexte aux exemples.	9 mai 2025

Ajout d'informations sur le type de manifeste DASH	Ajout d'explications sur la structure et les composants du MPD pour fournir un contexte aux exemples.	9 mai 2025
Ajout d'un préchargement récurrent	Sections mises à jour pour inclure des informations sur les planifications de prélecture récurrentes.	28 avril 2025
Ajout d'informations de filtrage des journaux par configuration	Ajout d'informations sur le filtrage des MediaTailor journaux émis pour une configuration.	31 mars 2025
Ajout d'informations sur les journaux distribués	Ajout d'une nouvelle section sur l'utilisation des journaux distribués pour publier des journaux qui MediaTailor émettent...	4 février 2025
Ajout d'un aperçu des publicités préconditionnées	Ajout d'une nouvelle section qui décrit comment MediaTailor gérer l'insertion d'annonces lorsque vous utilisez des publicités préconditionnées.	30 janvier 2025
Ajout d'un conditionnement publicitaire	Ajout du paramètre de conditionnement des fichiers multimédia en streaming.	30 janvier 2025
Guide de configuration SGAI Live	Ajout de conseils de configuration pour l'insertion d'annonces guidées par le serveur avec du contenu en direct.	15 janvier 2025

Types de journaux ajoutés	Ajout d'une nouvelle section répertoriant les types de journaux MediaTailor émis.	15 janvier 2025
Ajout d'informations de filtrage des journaux	Ajout d'une nouvelle section sur le filtrage des journaux qui MediaTailor émettent.	15 janvier 2025
Ajout du formatage des paramètres de requête pris en charge	Ajout de sections pour le formatage des paramètres de requête du manifeste et des paramètres de requête ADS.	2 janvier 2025
Informations d'intégration pour Google Ad Manager	Ajout d'une section sur l'intégration MediaTailor à Google Ad Manager côté serveur et côté client.	25 novembre 2024
Variable mise à jour	<code>breakabilityStartTime</code> Changé en <code>availabilityStartTime</code> .	6 mai 2024
Détails de personnalisation ajoutés	Ajout du détail Insertion Mode de personnalisation.	6 mai 2024
Règles du programme	Ajout d'un nouveau contenu sur les règles du programme.	25 avril 2024
Contenu mis à jour sur la création de chaînes	Ajout d'informations sur les règles du programme lors de la création de chaînes.	20 avril 2024
Contenu mis à jour sur l'ajout de programmes	Ajout d'informations sur les règles du programme lors de l'ajout de programmes.	20 avril 2024

moniteur de flux de travail	Analysez AWS les services multimédia et créez des cartes de signaux, des visualisations du flux de travail multimédia, entre ces services. Utilisez les cartes de signaux pour générer des alarmes et des notifications de surveillance à l'aide de CloudWatch EventBridge, et CloudFormation.	11 avril 2024
AlternateMedia et le journal As Run	Ajout d'une note sur l'AlternateMedia impact du journal As Run.	28 février 2024
Paramètres du manifeste mis à jour	Ajout d'informations sur les balises passthrough, en fonction du type de balisage publicitaire choisi.	28 novembre 2023
SCTE-35 messages pour les pauses publicitaires	Ajout d'informations sur les SCTE-35 balises insérées pour les types de balisage DateRange par rapport aux types de balisage Scte35 Enhanced publicitaire.	28 novembre 2023
Key-value paires pour le type de balisage Enhanced Scte35 publicitaire	Ajout d'informations sur la façon dont MediaTailor les paires clé-valeur soumises sont gérées pour le type de balisage Enhanced Scte35 publicitaire.	28 novembre 2023

Sources et opportunités de VOD	MediaTailor peut désormais détecter automatiquement les opportunités publicitaires sur les sources de VOD.	6 octobre 2023
Nouveau type d'authentification Sigv4 à détection automatique	MediaTailor prend désormais en charge le type AUTODETECT_SIGV4 d'accès.	18 août 2023
Contenu de suivi côté client mis à jour	Le contenu du suivi côté client a été mis à jour pour inclure des informations supplémentaires.	12 août 2023
Configuration MediaTailor et MediaPackage flux de travail pour utiliser les sources en direct	Ajout d'informations sur la configuration, les exigences générales et le comportement lorsque MediaTailor et les AWS Elemental MediaPackage flux de travail utilisent des sources en direct.	24 mai 2023
Documentation sur les paramètres de requête du manifeste	Ajout d'une section qui décrit les paramètres de requête du manifeste.	26 avril 2023
Documentation sur les publicités superposées	Ajout d'une section qui décrit les publicités superposées.	24 avril 2023
Décoration de l'identifiant publicitaire dans la documentation des manifestes	Ajout d'une section qui décrit la décoration des identifiants publicitaires dans les manifestes.	24 avril 2023

Ajout du mode de suppression AFTER_LIVE_EDGE	AFTER_LIVE_EDGE le mode de suppression des publicités est désormais disponible en plus du BEFORE_LIVE_EDGE mode.	21 février 2023
Nouveau journal As Run	Nouvelle rubrique sur le journal As Run.	19 janvier 2023
Mises à jour des meilleures pratiques IAM	Mise à jour du guide s'aligner sur les bonnes pratiques IAM. Pour plus d'informations, consultez Bonnes pratiques de sécurité dans IAM .	27 décembre 2022
Mises à jour des meilleures pratiques IAM	Mise à jour du guide s'aligner sur les bonnes pratiques IAM. Pour plus d'informations, consultez Bonnes pratiques de sécurité dans IAM .	27 décembre 2022
Nouvelles variables de données privées UPID de segmentation	MediaTailor prend désormais en charge des variables de session supplémentaires <code>scte.segmentation_upid.private_data.{index}</code> qui séparent les données privées séparées par des deux-points de la segmentation UPID de type 12 (MPU).	22 décembre 2022
Contenu des quotas mis à jour	Informations sur les quotas mises à jour et réorganisées.	13 septembre 2022

Quotas d'assemblage de chaînes ajoutés et corrigés	Ajout de quotas pour les sources en direct, les configurations de diffusion de segments, les demandes de manifeste et les transactions de canaux dans le service d'assemblage de MediaTailor canaux.	13 septembre 2022
Quotas d'assemblage de chaînes ajoutés et corrigés	Ajout de quotas pour la journalisation, les chaînes, les sources en direct, les programmes, les emplacements des sources et les politiques des chaînes dans le service d'assemblage de MediaTailor chaînes.	11 septembre 2022
Tableau des alertes relatives aux nouveaux assemblages de canaux	Vous pouvez maintenant voir un tableau qui explique les alertes d'assemblage de canaux.	1er septembre 2022
Nouvelles CloudWatch statistiques Amazon	De nouvelles CloudWatch mesures ont été ajoutées.	26 juin 2022
Sujet des appels publicitaires	Ajout d'un lien vers la prise en charge de la prélecture pour les réponses VAST.	25 mai 2022

[Nouvelles variables de requête ADS](#)

MediaTailor prend désormais en charge ces SCTE-35 variables supplémentaires dans les requêtes ADS :

```
scte.segmentation_type_id
scte.available_segments_expected
scte.delivery_not_restricted_flag
scte.segment_number
scte.segments_expected
scte.sub_segments_expected
scte.device_restrictions
scte.no_regional_blackout_flag
scte.archive_allowed_flag
scte.segmentation_event_id
```

27 avril 2022

[Nouveau sujet de politique de gestion des politiques IAM](#)

Ajout de deux nouvelles politiques gérées pour MediaTailor.

24 novembre 2021

[Nouvelle politique gérée par AWSElementalMediaTailorReadOnly](#)

Ajout d'une nouvelle politique AWS gérée qui accorde des autorisations permettant un accès en lecture seule aux MediaTailor ressources.

10 novembre 2021

[Nouvelle politique gérée par AWSElementalMediaTailorFullAccess](#)

Ajout d'une nouvelle politique AWS gérée qui permet un accès complet aux MediaTailor ressources.

10 novembre 2021

<u>Nouveau sujet adjoint confus</u>	Ajout d'une rubrique qui explique comment éviter le problème de confusion entre les sous-ministres.	4 novembre 2021
<u>Sujet relatif à la prélecture des annonces</u>	MediaTailor peut désormais prévisualiser les publicités pour les pauses publicitaires avant qu'elles ne se produisent.	12 octobre 2021
<u>Ajout de paramètres de configuration de journalisation pour les configurations de lecture</u>	Utilisez les paramètres de configuration de journalisation pour contrôler les paramètres liés aux journaux de configuration de lecture.	28 septembre 2021
<u>SCTE-35 messages pour les pauses publicitaires</u>	Ajout d'informations sur l'utilisation <code>segmentation_descriptor</code> des messages avec <code>time_signal</code> les messages.	1er septembre 2021
<u>Nouveau mode de lecture linéaire</u>	Ajout d'un nouveau mode de lecture linéaire.	1er septembre 2021
<u>Nouveau type de transition absolue</u>	Ajout de la prise en charge des types de transition absolus, que vous pouvez utiliser pour définir l'heure de début d'une horloge murale pour votre programme sur des canaux linéaires.	1er septembre 2021

<u>Sujet des alertes relatives à l'assemblage de nouveaux canaux</u>	Vous pouvez désormais surveiller vos ressources d'assemblage de canaux à l'aide d' MediaTailor alertes. Lorsqu'un problème ou un problème potentiel survient avec vos ressources d'assemblage de canaux, MediaTailor génère des alertes.	14 juillet 2021
<u>Quotas d'assemblage de canaux corrigés pour les demandes de sortie de canaux</u>	Quotas corrigés pour les demandes de sortie de canaux dans le service d'assemblage de MediaTailor canaux.	29 juin 2021
<u>Nouveau type d'authentification pour l'emplacement de la source</u>	MediaTailor prend désormais en charge l'authentification par jeton d'accès à Secrets Manager.	16 juin 2021
<u>Informations sur les nouveaux niveaux</u>	Ajout d'informations sur les modes et les types de sources pris en charge par chaque niveau.	13 juin 2021
<u>Nouvelles informations sur le type de source</u>	Pour les chaînes standard, ajout d'informations sur le type de source diffusé par le programme.	13 juin 2021

[Nouvelle documentation sur les sources MediaTailor en direct](#)

Une source en direct représente un flux en direct unique que vous ajoutez à votre localisation source. Après avoir créé votre chaîne, vous pouvez ajouter des sources en direct à votre emplacement source et associer chaque source en direct à un programme.

13 juin 2021

[Prise en charge de types d'UPID supplémentaires](#)

MediaTailor prend désormais en charge les types UPID de segmentation ADS Information (0xE) et définis par l'utilisateur (0x1).

15 avril 2021

[Nouvelles variables dynamiques UPID de segmentation](#)

Il existe trois nouvelles variables dynamique :

```
scte.segm
entation_upid.asse
tId scte.segm
entation_upid.cueD
ata.key , etscte.segm
entation_upid.cueD
ata.value . Ces variables
```

sont utilisées conjointement avec le type UPID de segmentation MPU (0xC) pour les flux de travail Podbuster.

15 avril 2021

[Description du nouveau service d'assemblage de canaux](#)

Ajout d'informations sur le nouveau service d'assemblage de canaux.

11 mars 2021

Nouvelle documentation du service d'assemblage de MediaTailor canaux	Channel Assembly est un nouveau service de manifeste uniquement qui vous permet de créer des chaînes de streaming linéaires à partir de votre contenu de vidéo à la demande (VOD) existant.	11 mars 2021
Ajout de quotas d'assemblage de chaînes	Ajout de quotas pour le nouveau service d'assemblage de MediaTailor chaînes.	11 mars 2021
Nouvelles conditions d'assemblage des chaînes	Ajout de termes correspondant au nouveau service d'assemblage de chaînes.	10 mars 2021
Support pour le marquage des assemblages de canaux	Ajout de la prise en charge du balisage des ressources d'assemblage de canaux dans AWS Elemental MediaTailor. Chaînes SourceLocations et balisage de VodSources support.	9 mars 2021
Nouveau sujet sur les variables dynamiques	MediaTailor prend désormais en charge les variables de domaine dynamiques.	25 février 2021
Ajout de paramètres d'alias de configuration facultatifs	Utilisez des alias de configuration en plus des variables de domaine pour configurer dynamiquement les domaines lors de l'initialisation de la session.	25 février 2021

<u>Nouvelle variable publicitaire scte.segmentation_upid dynamique</u>	Ajout de la prise en charge de la variable publicitaire dynamique des données de scte.segmentation_upid session.	5 décembre 2020
<u>Nouvelle rubrique dédiée aux marqueurs publicitaires</u>	Le transfert des marqueurs publicitaires est désormais disponible pour les manifestes HLS.	29 octobre 2020
<u>Paramètres avancés de configuration mis à jour</u>	Le passthrough des marqueurs publicitaires est un nouveau paramètre avancé de configuration de lecture.	14 octobre 2020
<u>Nouveau mode journal de débogage</u>	Nouvelle rubrique sur le mode journal DEBUG.	14 août 2020
<u>Clarification concernant l'attribut de EXT-X-CUE-OUT durée pour les pare-chocs</u>	Mise à jour des exigences relatives aux pare-chocs afin que, pour HLS, l'attribut soit obligatoire pour chaque EXT-X-CUE-OUT étiquette.	5 août 2020
<u>Paramètre du mode de rapport d'initialisation de session</u>	Ajout de la prise en charge du reportingMode paramètre dans les demandes d'initialisation de session. Vous pouvez désormais spécifier si MediaTailor le suivi des publicités est effectué côté client ou côté serveur lors de l'initialisation d'une session.	30 juillet 2020
<u>Nouveau thème sur les pare-chocs</u>	Ajout d'un nouveau sujet sur les pare-chocs	27 juillet 2020

La suppression des publicités est disponible pour DASH	La suppression des publicités est désormais disponible pour DASH. Suppression de la restriction HLS-only « » de la rubrique sur la suppression des publicités.	3 juin 2020
Mettre à jour les noms spécifiques à la console	Mise à jour des noms spécifiques à la console pour refléter une version plus récente de l'interface utilisateur de la console.	1er mai 2020
Nouvelle variable publicitaire avail.index dynamique	Ajout de la prise en charge de la nouvelle variable publicitaire dynamique de données de session <code>avail.index</code> .	13 mars 2020
AdVerifications Nouveautés et Extensions éléments	Pour les rapports côté client, les éléments Extensions et AdVerifications sont pris en charge.	10 mars 2020
Configuration du seuil de personnalisation	Ajout de la prise en charge de la configuration facultative du seuil de personnalisation.	14 février 2020
Manifestes DASH VOD	Ajout de la prise en charge des manifestes DASH de vidéo à la demande (VOD) à partir du serveur d'origine , avec sortie de manifeste à plusieurs périodes.	23 décembre 2019
Support de console pour le nom du profil de transcodage	Ajout d'une description pour le nom du profil de transcodage dans la configuration.	23 décembre 2019

<u>Tableaux de limites mis à jour</u>	Mise à jour des limites pour les redirections ADS et les délais d'expiration ADS.	18 décembre 2019
<u>Documentez en temps réel les comportements avant le lancement</u>	Ajout de Pre-Roll la section Insertion d'annonces pour décrire le fonctionnement des publicités prédiffusées en direct. AWS Elemental MediaTailor	26 novembre 2019
<u>Prise en charge des publicités en direct avant diffusion</u>	Ajout de la prise en charge de l'insertion de publicités preroll au début d'un flux en direct.	11 septembre 2019
<u>Analyse des journaux ADS dans Amazon CloudWatch Logs Insights</u>	Ajout d'informations sur l'utilisation AWS Elemental MediaTailor des journaux ADS et CloudWatch Logs Insights pour analyser vos MediaTailor sessions.	13 août 2019
<u>Nouveau chapitre sur la sécurité</u>	Ajout d'un chapitre sur la sécurité pour améliorer et standardiser la couverture.	23 mai 2019
<u>Manifestes DASH à période unique</u>	Ajout de la prise en charge des manifestes DASH à période unique à partir du serveur d'origine, avec sortie de manifeste à plusieurs périodes.	4 avril 2019

Prise en charge des SCTE-35 UPID dans l'URL ADS	Ajout de la prise en charge pour l'inclusion d'un ID de programme unique (UPID) dans l'URL ADS (Ad decision server). Cela permet à l'ADS de fournir un ciblage publicitaire au niveau du programme dans un flux linéaire en direct.	28 mars 2019
Client-side les rapports prennent en charge les publicités associées	Pour les rapports côté client, la réponse à l'URL AWS Elemental MediaTailor de suivi inclut désormais les métadonnées des annonces associées.	28 mars 2019
Documentation relative aux marqueurs publicitaires HLS	Ajout d'une section qui décrit la prise en charge des marqueurs de publicité HLS.	1 mars 2019
Support de balisage	Ajout de la prise en charge du balisage des ressources de configuration dans AWS Elemental MediaTailor. Le balisage vous permet d'identifier et d'organiser vos AWS ressources, de contrôler l'accès à celles-ci et de suivre vos AWS coûts.	14 février 2019
Informations de AWS CloudTrail journalisation ajoutées	Ajout d'une rubrique sur l'utilisation CloudTrail de la journalisation des actions dans l' AWS Elemental MediaTailor API.	11 février 2019

Ajout d'une section sur les erreurs de lecture	Ajout d'informations sur les erreurs qui peuvent être renvoyés par MediaTailor pendant la lecture, en réponse à des demandes provenant d'un lecteur ou d'un réseau de diffusion de contenu (CDN).	4 février 2019
Binaire codé DASH en base64	Ajout de la prise en charge de la fourniture des informations de jointure dans les manifestes dans le binaire encodé en base64, à l'intérieur des marqueurs <scte35:Signal> <scte35:Binary> .	4 janvier 2019
Signal horaire DASH	Ajout de la prise en charge de la fourniture des informations de jointure dans les manifestes à l'intérieur des marqueurs <scte35:TimeSignal> .	5 décembre 2018
Support de localisation DASH	Ajout du support pour le MPEG-DASH <Location> tag.	4 décembre 2018
Assistance DASH	Ajout du support pour les MPEG-DASH manifestes.	14 novembre 2018
Tableaux de limites mis à jour	Mise à jour des limites pour les configurations et la taille des manifestes.	13 octobre 2018

Statistiques nouvelles et mises à jour	Ajout de métriques pour le serveur ADS (Ad decision server) et de délais d'expiration d'origine, et mise à jour de l'ADS et des définitions d'erreurs d'origine pour inclure les réponses de dépassement du délai d'attente.	13 octobre 2018
Meilleure couverture documentaire pour les cas d'utilisation d'insertion d'annonces côté serveur et côté client	Description étendue et exemples de couverture de l'utilisation de variables de publicité dynamiques pour l'insertion publicitaire côté serveur et l'insertion publicitaire côté client.	1 octobre 2018
Nouvelles régions	Ajout de la prise en charge des régions PDX et FRA.	18 juillet 2018
VAST/VPAID	Ajout d'informations sur VAST et VPAID.	16 mars 2018
CloudWatch	Ajout d'informations sur CloudWatch les métriques, les espaces de noms et les dimensions disponibles.	16 mars 2018
Nouvelles régions	Ajout de la prise en charge des régions Asie-Pacifique (Singapour), Asie-Pacifique (Sydney) et Asie-Pacifique (Tokyo).	8 février 2018

[Informations sur la politique IAM](#)

Ajout d'informations de politique IAM spécifiques à AWS Elemental MediaTailor. Ajout d'instructions pour la création de rôles non administrateurs avec des autorisations limitées.

3 janvier 2018

[Première publication](#)

Première version de la documentation.

27 novembre 2017

 Note

- Les services AWS multimédia ne sont pas conçus ni destinés à être utilisés avec des applications ou dans des situations nécessitant des performances fiables, telles que les opérations de sécurité des personnes, les systèmes de navigation ou de communication, le contrôle du trafic aérien ou les appareils de survie dans lesquels l'indisponibilité, l'interruption ou la défaillance des services pourraient entraîner la mort, des blessures corporelles, des dommages matériels ou des dommages environnementaux.

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.