



ユーザーガイド

AWS Deadline クラウド



Version latest

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

AWS Deadline クラウド: ユーザーガイド

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

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Deadline Cloud とは	1
Deadline Cloud の機能	1
概念と用語	2
Farm リソース	2
ジョブ実行リソース	3
その他の重要な概念と用語	5
Deadline Cloud の開始方法	7
Deadline Cloud へのアクセス	7
関連サービス	7
Deadline Cloud の仕組み	8
Deadline Cloud のアクセス許可	9
Deadline Cloud でのソフトウェアサポート	10
パイプライン統合	10
パイプライン統合とは	10
ファームが のオンプレミススタジオの例 AWS	11
開始方法	14
をセットアップする AWS アカウント	14
にサインアップする AWS アカウント	14
ファームインフラストラクチャをセットアップする	14
モニターを作成する	15
ファームの詳細を定義する	17
キューの詳細を定義する	18
フリートの詳細を定義	19
確認と作成	20
ワークステーションのセットアップ	20
ステップ 1: Deadline Cloud 送信者をインストールする	21
ステップ 2: Deadline Cloud モニターをインストールしてセットアップする	25
ステップ 3: Deadline Cloud 送信者を起動する	30
モニターの使用	31
追加のリージョンでモニターを作成する	32
クロスリージョン IAM アイデンティティセンターへのアクセス	33
IAM Identity Center マルチリージョンレプリケーション	34
Deadline Cloud モニター URL を共有する	34
Deadline Cloud モニターを開く	35

言語設定を変更する	36
ジョブバンドルを送信する	37
キューとフリートの詳細を表示する	37
ジョブ、ステップ、タスクを管理する	38
ジョブの詳細を表示する	39
ジョブをアーカイブする	40
ジョブをキューに入れる	41
ジョブを再送信する	41
ステップを表示する	41
タスクを表示する	42
セッションログとワーカーログを表示する	43
ワーカーダッシュボードを表示する	44
ユースケース	45
完了した出力をダウンロードする	47
ジョブアタッチメントを参照する	48
アタッチメントブラウザを開く	48
ファイルのナビゲーションとフィルタリング	49
ファイルのダウンロード	49
ファイルのプレビュー	50
デスクトップのデプロイとワークフローを自動化する	50
Deadline Cloud Monitor 実行可能ファイルの検索	51
ユーザーアクセスを合理化するためのプロファイルの設定	51
Deadline Cloud モニターをワークフローに統合する	52
ファーム	55
ファームを作成する	55
[キュー]	56
キューを作成する	56
キュー環境を作成する	58
デフォルトのcondaキュー環境	58
キューとフリートを関連付ける	62
キューフリートの関連付けを停止する	63
キューフリートの関連付けを再アクティブ化する	64
フリート	65
サービスマネージドフリート	65
SMF を作成する	66
GPU アクセラレーターを使用する	68

永続ストレージ	69
ソフトウェアライセンス	74
VFX プラットフォーム	75
AMI ソフトウェアの内容	76
カスタマーマネージドフリート	80
自動スケーリング設定	80
スケールアウトレート	81
ワーカーのアイドル時間	81
スタンバイワーカー数	82
自動スケーリング設定の構成	84
ユーザーの管理	85
ID ソースについて	85
を使用してユーザーを作成する IAM アイデンティティセンターディレクトリ	86
外部 IdP を使用してユーザーを管理する	88
モニターアクセスの制限	88
アクセスレベルについて	89
アクセスレベルのアクセス許可マトリックス	90
メンバーシップの継承	91
アクセス許可の割り当て	93
ジョブ	95
送信者の使用	96
共有ジョブ設定タブ	98
ジョブ固有の設定タブ	100
ジョブアタッチメントタブ	101
ホスト要件タブ	103
処理ジョブ	104
ジョブのモニタリング	105
対応ソフトウェア	108
Adobe After Effects	108
サポートの概要	109
After Effects バージョンの互換性	109
Deadline Cloud Conda チャンネル	109
開始方法	110
インストール	110
After Effects 送信者の使用	114
トラブルシューティング	118

詳細設定	123
オープンソースリソース	123
Autodesk 3ds Max	123
サポートの概要	124
3ds Max バージョンの互換性	124
3ds 他のデジタルコンテンツ作成ツールとの最大の違い	124
開始方法	125
フリートホスト設定	125
インストール	126
Autodesk 3ds Max 送信者の使用	126
V-Ray スタンドアロンタイルレンダリング	133
詳細設定	136
3ds Max レンダラー	136
オープンソースリソース	136
Autodesk Maya	137
サポートの概要	137
Maya バージョンの互換性	137
Deadline Cloud Conda チャンネル	138
開始方法	139
インストール	139
Autodesk Maya 送信者の使用	140
詳細設定	144
Maya レンダリングエンジン	145
Maya プラグイン	146
オープンソースリソース	147
Autodesk VRED	147
サポートの概要	147
VRED バージョンの互換性	147
Deadline Cloud Conda チャンネル	148
要件	148
開始方法	149
インストール	149
Autodesk VRED 送信者の使用	150
詳細設定	157
VRED レンダリングエンジン	157
オープンソースリソース	157

ブレンダー	158
サポートの概要	158
Blender バージョンの互換性	158
Deadline Cloud Conda チャンネル	159
開始方法	160
インストール	160
Blender 送信者の使用	163
詳細設定	166
Blender レンダーエンジン	167
オープンソースリソース	167
エピック Unreal エンジン	168
サポートの概要	168
Unreal Engine バージョンの互換性	168
Deadline Cloud Conda チャンネル	169
開始方法	169
送信者のインストール	169
フリートの作成	175
テストレンダーの送信	176
カスタマーマネージドフリート (CMF) ワーカーのセットアップ	177
認証情報管理を実施する	181
Perforce レンダリングジョブの作成	188
カスタムホストの要件	198
詳細設定	202
Unreal Engine レンダリング機能	202
オープンソースリソース	203
Foundry Nuke	203
サポートの概要	203
Nuke バージョンの互換性	204
Deadline Cloud Conda チャンネル	204
開始方法	205
インストール	205
Nuke 送信者の使用	207
詳細設定	213
Nuke 合成機能	214
オープンソースリソース	214
KeyShot Studio	215

サポートの概要	215
KeyShot バージョンの互換性	215
前提条件	215
開始方法	216
インストール	216
KeyShot 送信者の使用	217
詳細設定	222
オープンソースリソース	222
Maxon シネマ 4D	222
サポートの概要	223
Cinema 4D バージョンの互換性	223
Deadline Cloud Conda チャンネル	224
開始方法	225
クイックスタート	226
送信者機能	230
トラブルシューティング	237
サポート情報	240
詳細設定	244
Cinema 4D プラグイン	245
オープンソースリソース	247
SideFX Houdini	247
サポートの概要	247
Houdini バージョンの互換性	247
Deadline Cloud Conda チャンネル	248
開始方法	249
インストール	249
Houdini 送信者の使用	250
トラブルシューティング	256
詳細設定	256
Houdini レンダーエンジン	257
オープンソースリソース	258
Storage	259
ストレージプロファイル	260
共有ファイルシステムの場合	262
ジョブアタッチメントの場合	263
ジョブアタッチメント	264

ジョブアタッチメント S3 バケットの暗号化	265
ジョブアタッチメントバケットを置き換える	266
S3 バケットでのジョブアタッチメントの管理	267
仮想ファイルシステム	268
自動ダウンロード	270
支出と使用状況を追跡する	289
コストの前提	289
コストスケール係数	290
コストスケール係数値	291
コストスケール係数を設定する	291
コストスケール係数がコストツールに与える影響	291
予算によるコストの管理	292
前提条件	292
Deadline Cloud 予算マネージャーを開く	292
予算を作成する	293
予算を表示する	294
予算を編集する	294
予算を非アクティブ化する	295
EventBridge イベントで予算をモニタリングする	295
使用状況とコストを追跡する	296
前提条件	297
使用量エクスペローラーを開く	297
使用量エクスペローラーを使用する	296
コスト管理	300
コスト管理のベストプラクティス	301
セキュリティ	304
データ保護	305
保管中の暗号化	306
転送中の暗号化	306
キー管理	307
ネットワーク間トラフィックのプライバシー	317
オプトアウト	317
Identity and Access Management	318
オーディエンス	319
アイデンティティを使用した認証	319
ポリシーを使用したアクセスの管理	321

Deadline Cloud と IAM の連携方法	322
アイデンティティベースのポリシーの例	328
AWS マネージドポリシー	338
サービスロール	342
トラブルシューティング	356
コンプライアンス検証	358
耐障害性	358
インフラストラクチャセキュリティ	359
設定と脆弱性の分析	359
サービス間での不分別な代理処理の防止	360
AWS PrivateLink	361
考慮事項	362
Deadline Cloud エンドポイント	362
エンドポイントの作成	363
制限されたネットワーク環境	364
AWS 許可リストの API エンドポイント	364
許可リストのウェブドメイン	365
許可リストを作成する環境固有のエンドポイント	365
セキュリティのベストプラクティス	366
データ保護	366
IAM アクセス許可	367
ユーザーおよびグループとしてジョブを実行する	367
ネットワーク	368
ジョブデータ	368
ファーム構造	369
ジョブアタッチメントキュー	369
カスタムソフトウェアバケット	372
ワーカーホスト	373
ホスト設定スクリプト	374
ワークステーション	374
ダウンロードしたソフトウェアを検証する	375
Deadline クラウドアシスタント	383
アシスタントの仕組み	383
重要な考慮事項	384
Deadline Cloud Assistant の有効化	385
前提条件	385

アシスタントの有効化	385
アシスタントの無効化	385
必要なアクセス許可	386
IAM ポリシー	386
クロスリージョン推論	387
セキュリティ	388
モデル情報	389
データプライバシー	389
ネットワークパス	390
組織レベルのコントロール	390
監査証跡	391
不正検出	391
フィードバックデータ	391
コスト	392
アシスタントのコストの追跡	392
サービスクォータとスロットリング	394
クォータ引き上げのリクエスト	394
クォータ使用状況のモニタリング	395
トラブルシューティング	395
Assistant がエラーを返すか、応答が遅い	395
Assistant が Amazon CloudWatch logsにアクセスできない	395
その他のリソース	396
モニタリング	397
クォータ	399
AWS CloudFormation リソース	405
Deadline Cloud と CloudFormation テンプレート	405
の詳細 CloudFormation	405
トラブルシューティング	406
ユーザーがファーム、フリート、またはキューを表示できないのはなぜですか?	406
ユーザーアクセス	406
ワーカーがジョブを取得しないのはなぜですか?	407
フリートロールの設定	407
ワーカーが停止しているのはなぜですか?	408
OpenJD 環境からのワーカーの停止	408
ジョブのトラブルシューティング	409
ジョブの作成が失敗したのはなぜですか?	409

ジョブに互換性がないのはなぜですか?	409
ジョブの準備が整うのはなぜですか?	410
ジョブが失敗したのはなぜですか?	410
ステップが保留になっているのはなぜですか?	411
Deadline Cloud モニターデスクトップアプリケーションログ	411
その他のリソース	411
リリースノート	412
AWS 用語集	434
.....	cdxxxv

AWSDeadline Cloud とは

Deadline Cloud は、デジタルコンテンツ作成パイプラインとワークステーションから直接 Amazon Elastic Compute Cloud (Amazon EC2) インスタンスでレンダリングプロジェクトとジョブを作成および管理するためにAWS のサービス使用できる です。

Deadline Cloud は、コンソールインターフェイス、ローカルアプリケーション、コマンドライン ツール、および API を提供します。Deadline Cloud を使用すると、ファーム、フリート、ジョブ、ユーザーグループ、ストレージを作成、管理、モニタリングできます。また、ハードウェア機能を指定し、特定のワークロードの環境を作成し、本番稼働に必要なコンテンツ作成ツールを Deadline Cloud パイプラインに統合することもできます。

Deadline Cloud は、すべてのレンダリングプロジェクトを 1 か所で管理するための統合インターフェイスを提供します。ユーザーを管理し、プロジェクトを割り当て、ジョブロールのアクセス許可を付与できます。

トピック

- [Deadline Cloud の機能](#)
- [Deadline Cloud の概念と用語](#)
- [Deadline Cloud の開始方法](#)
- [Deadline Cloud へのアクセス](#)
- [関連サービス](#)
- [Deadline Cloud の仕組み](#)
- [Deadline Cloud をパイプラインに統合する](#)

Deadline Cloud の機能

Deadline Cloud がビジュアルコンピューティングワークロードの実行と管理に役立つ主な方法をいくつか紹介します。

- ファーム、キュー、フリートをすばやく作成します。ステータスをモニタリングし、ファームとジョブのオペレーションに関するインサイトを取得します。
- Deadline Cloud ユーザーとグループを一元管理し、アクセス許可を割り当てます。
- を使用して、プロジェクトユーザーと外部 ID プロバイダーのサインインセキュリティを管理しますAWS IAM アイデンティティセンター。

- AWS Identity and Access Management(IAM) ポリシーとロールを使用して、プロジェクトリソースへのアクセスを安全に管理します。
- タグを使用してプロジェクトリソースを整理し、すばやく検索します。
- プロジェクトのプロジェクトリソースの使用状況と推定コストを管理します。
- クラウドまたは対面でのレンダリングをサポートするために、幅広いコンピューティング管理オプションを提供します。

Deadline Cloud の概念と用語

Deadline Cloud AWS の開始に役立つように、このトピックではその主要な概念と用語の一部について説明します。

Farm リソース

この図は、Deadline Cloud ファームリソースがどのように連携するかを示しています。

ファーム

ファームには、ジョブの送信と実行に関連する他のすべてのリソースが含まれています。ファームは互いに独立しているため、本番環境の分離に役立ちます。

キュー

キューは、関連するフリートでスケジューリングするためのジョブを保持します。ユーザーはジョブをキューに送信し、キュー内の優先度とステータスを管理できます。ジョブを実行するには、キューをキューとフリートの関連付けを持つフリートに関連付ける必要があり、キューを複数のフリートに関連付けることができます。

フリート

フリートには、実行中のジョブのコンピューティング容量が含まれています。フリートは、サービス管理でもカスタマー管理でもかまいません。サービスマネージドフリートは Deadline Cloud で実行され、自動スケーリング、ライセンス、ソフトウェアアクセスなどの組み込み機能が含まれています。カスターマネージドフリートは、Amazon EC2 インスタンスやオンプレミスサーバーなどの独自のコンピューティングリソースで実行されます。

予算

予算は、ジョブアクティビティの支出しきい値を設定し、しきい値に達したときにジョブのスケジューリングの停止などのアクションを実行できます。

キュー環境

キュー環境は、ワークロード環境をセットアップまたは削除するために各ワーカーで実行されるスクリプトを定義します。環境変数の設定、ソフトウェアのインストール、アセットストレージの設定に役立ちます。

ストレージプロファイル

ストレージプロファイルは、ホストとワークステーションのグループの設定であり、ファイルシステム上のデータがどこにあるかを示します。Deadline Cloud は、 から送信Windowsされ、 で実行されるジョブなど、異なる設定のホストでジョブを実行するときに、ストレージプロファイルを使用してパスをマッピングしますLinux。

[制限]

制限により、フローティングライセンスなどの共有リソースの使用状況を追跡し、ジョブ間での割り当て方法を制御できます。制限は、キューと制限の関連付けを持つキューに関連付けられません。

モニタリング

モニターは Deadline Cloud Monitor ウェブアプリケーションの URL を設定し、エンドユーザーがジョブをモニタリングおよび管理できるようにします。ブラウザまたは Deadline Cloud Monitor デスクトップアプリケーションからアクセスできます。

ジョブ実行リソース

この図は、Deadline Cloud ジョブリソースがどのように連携するかを示しています。

ジョブ

ジョブは、ユーザーが Deadline Cloud に送信して、使用可能なワーカーでスケジュールおよび実行するための一連の作業です。ジョブは 3D シーンをレンダリングしたり、シミュレーションを実行したりできます。ジョブは、ランタイム環境とプロセス、およびジョブ固有のパラメータを定義する再利用可能なジョブテンプレートから作成されます。ジョブには、実行する作業を定義するステップとタスクが含まれており、優先順位、最大ワーカー数、再試行設定で設定できます。

ジョブの優先度

ジョブの優先度は、Deadline Cloud がキュー内のジョブを処理するおおよその順序です。ジョブの優先度は 1~100 の間で設定できます。優先度の高いジョブは通常、最初に処理されます。優先度が同じジョブは、受信した順序で処理されます。

ジョブプロパティ

ジョブプロパティは、レンダリングジョブを送信するときに定義する設定です。例としては、フレーム範囲、出力パス、ジョブアタッチメント、レンダリング可能なカメラなどがあります。プロパティは、レンダリングの送信元の DCC によって異なります。

Step

ステップは、タスクパラメータ値を除いて、多くの同じタスクを実行するためのテンプレートを提供するジョブの一部です。ステップには他のステップへの依存関係があるため、シーケンシャル実行パスまたは並列実行パスを使用して複雑なワークフローを作成できます。レンダリングジョブでは、ステップは多くの場合、フレームをレンダリングするためのコマンドを定義し、フレーム番号をタスクパラメータとして使用します。

タスク

タスクは、Deadline Cloud での作業の最小単位です。タスクはステップの一部であり、ワーカーによって実行され、ジョブの一部として実行する必要がある個々のオペレーションを表します。タスクは特定のパラメータで設定でき、その機能と可用性に基づいてワーカーに割り当てられます。レンダリングジョブでは、タスクが 1 つのフレームをレンダリングすることがよくあります。

ワーカー

ワーカーはフリートの一部であり、ジョブからタスクを実行します。ワーカーは、GPU アクセラレーター、CPU アーキテクチャ、オペレーティングシステムなどの特定の機能で設定できます。サービスマネージドフリートでは、フリートがスケールアウトおよびスケールインすると、ワーカーが自動的に作成されます。

インスタンス

フリートは CPU リソースにインスタンスを使用します。インスタンスは Amazon EC2 パフォーマンスインスタンスです。Deadline Cloud はオンデマンドインスタンスとスポットインスタンスを使用します。

オンデマンドインスタンス

オンデマンドインスタンスの料金は 2 番目で、長期的なコミットメントはなく、中断されません。

スポットインスタンス

スポットインスタンスは、割引価格で使用できる予約されていない容量ですが、オンデマンドリクエストによって中断される可能性があります。

待機して保存する

待機と保存機能を使用すると、ジョブのスケジュールが遅れて低コストになり、オンデマンドリクエストとスポットリクエストによって中断される可能性があります。Wait and Save は、Deadline Cloud のサービスマネージドフリートでのみ使用できます。

Wait and Save は、Deadline Cloud AWS でのビジュアルコンピューティングワークロードの実行を管理するためのものです。詳細については、[AWS「サービス条件」](#)を参照してください。

セッション

セッションは、ジョブに対するワーカーの作業のシーケンスを表します。1回のセッション中に、ワーカーに複数のタスクが割り当てられ、タスクが順番に実行されます。多くの場合、セッションには、タスクアクションを実行する前に環境を設定し、アセットをロードするセットアップアクションがあります。

セッションアクション

セッションアクションは、環境の設定、タスクの実行、アセットの同期など、セッション中に実行される特定のオペレーションを表します。

その他の重要な概念と用語

使用状況エクスペローラー

Usage Explorer は Deadline Cloud Monitor の機能です。コストと使用量のおおよその見積もりを提供します。

予算マネージャー

Budget Manager は Deadline Cloud モニターの一部です。予算マネージャーを使用して、予算を作成および管理します。また、これを使用して、予算内に収まるようにアクティビティを制限することもできます。

Deadline Cloud クライアントライブラリ

オープンソースのクライアントライブラリには、Deadline Cloud を管理するためのコマンドラインインターフェイスとライブラリが含まれています。機能には、Open Job Description 仕様に基

づくジョブバンドルの Deadline Cloud への送信、ジョブアタッチメント出力のダウンロード、コマンドラインインターフェイス (CLI) を使用したファームのモニタリングが含まれます。

デジタルコンテンツ作成アプリケーション (DCC)

デジタルコンテンツ作成アプリケーション (DCCs) は、デジタルコンテンツを作成するサードパーティー製品です。Deadline Cloud には、Autodesk Maya、Blender、Maxon Cinema 4D などの多くの DCCs との統合が組み込まれているため、DCC 内からジョブを送信し、事前設定されたソフトウェアとライセンスを使用してサービスマネージドフリートでレンダリングできます。

ジョブアタッチメント

ジョブアタッチメントは、テクスチャ、3D モデル、ライティングリグなどのジョブの一部としてアセットをアップロードおよびダウンロードする Deadline Cloud 機能です。ジョブアタッチメントは Amazon S3 に保存されるため、共有ネットワークストレージが不要になります。

ジョブテンプレート

ジョブテンプレートは、ランタイム環境と、Deadline Cloud ジョブの一部として実行されるすべてのプロセスを定義します。

Deadline Cloud 送信者

Deadline Cloud 送信者は、ユーザーが DCC 内からジョブを簡単に送信できるようにする DCC のプラグインです。

ライセンスエンドポイント

ライセンスエンドポイントは、サードパーティー製品の Deadline Cloud の使用ベースのライセンスを VPC 内で利用できるようにします。このモデルは従量制料金で、使用した時間と分数に対して課金されます。ライセンスエンドポイントはファームに接続されておらず、個別に使用できます。

[タグ]

タグは、AWS リソースに割り当てることができるラベルです。各タグは、お客様が定義するキーとオプション値で構成されています。タグを使用すると、目的、所有者、環境など、さまざまな方法で AWS リソースを分類できます。

使用量ベースのライセンス (UBL)

使用量ベースのライセンス (UBL) は、一部のサードパーティー製品で使用できるオンデマンドライセンスモデルです。このモデルは従量制料金で、使用した時間と分数に対して課金されます。

Deadline Cloud の開始方法

Deadline Cloud を使用すると、Amazon EC2 インスタンス設定や Amazon Simple Storage Service (Amazon S3) バケットなどのデフォルト設定とリソースを使用してレンダーファームをすばやく作成できます。

レンダーファームを作成するときに、設定とリソースを定義することもできます。このメソッドは、デフォルト設定とリソースを使用するよりも時間がかかりますが、より細かく制御できます。

Deadline Cloud [の概念と用語を理解したら](#)、「ファームの作成、ユーザーの追加、役立つ情報へのリンク」のstep-by-stepの手順については、「[開始方法](#)」を参照してください。

Deadline Cloud へのアクセス

Deadline Cloud には、次のいずれかの方法でアクセスできます。

- Deadline Cloud コンソール – ブラウザでコンソールにアクセスしてファームとそのリソースを作成し、ユーザーアクセスを管理します。詳細については、「[開始する](#)」を参照してください。
- Deadline Cloud Monitor – 優先順位やジョブステータスの更新など、レンダリングジョブを管理します。ファームをモニタリングし、ログとジョブのステータスを表示します。所有者のアクセス許可を持つユーザーの場合、Deadline Cloud モニターは使用状況を調べて予算を作成するためのアクセスも提供します。Deadline Cloud モニターは、ウェブブラウザとデスクトップアプリケーションの両方で使用できます。
- AWSSDK およびAWS CLI – AWS Command Line Interface(AWS CLI) を使用して、ローカルシステムのコマンドラインから Deadline Cloud API オペレーションを呼び出します。詳細については、「[デベロッパーワークステーションのセットアップ](#)」を参照してください。

関連サービス

Deadline Cloud は以下を使用しますAWS のサービス。

- Amazon CloudWatch – CloudWatch を使用すると、プロジェクトと関連するAWSリソースをモニタリングできます。詳細については、「Deadline [Cloud Developer Guide](#)」の「[Monitoring with CloudWatch](#)」を参照してください。
- Amazon EC2 – クラウドでアプリケーションを実行する仮想サーバーAWS のサービスを提供します。ワークロードに Amazon EC2 インスタンスを使用するようにプロジェクトを設定できます。詳細については、「[Amazon EC2 インスタンス](#)」を参照してください。

- Amazon EC2 Auto Scaling – Auto Scaling を使用すると、インスタンスの需要の変化に応じてインスタンス数を自動的に増減できます。Auto Scaling は、インスタンスが失敗した場合でも、必要な数のインスタンスを実行していることを確認するのに役立ちます。Deadline Cloud で Auto Scaling を有効にすると、Auto Scaling によって起動されたインスタンスはワークロードに自動的に登録されます。同様に、Auto Scaling によって終了したインスタンスは、ワークロードから自動的に登録解除されます。詳細については、[Amazon EC2 Auto Scaling ユーザーガイド](#)」を参照してください。
- AWS PrivateLink– は、トラフィックをパブリックインターネットに公開することなくAWS のサービス、仮想プライベートクラウド (VPCs) とオンプレミスネットワーク間のプライベート接続AWS PrivateLinkを提供します。AWS PrivateLinkを使用すると、さまざまなアカウントやVPCs。詳細については、「[AWS PrivateLink](#)」を参照してください。
- Amazon S3 – Amazon S3 はオブジェクトストレージサービスです。Deadline Cloud は Amazon S3 バケットを使用してジョブアタッチメントを保存します。詳細については、「[Amazon S3 ユーザーガイド](#)」を参照してください。
- IAM Identity Center – IAM Identity Center は、割り当てられたすべてのアカウントとアプリケーションへのシングルサインオンアクセスを 1 か所からユーザーに付与AWS のサービスできます。また、AWS Organizations のすべてのアカウントへのマルチアカウントアクセスとユーザーのアクセス許可を、一元的に管理することも可能です。詳細については、「[AWS IAM アイデンティティセンターに関するよくある質問](#)」を参照してください。

Deadline Cloud の仕組み

Deadline Cloud を使用すると、デジタルコンテンツ作成 (DCC) パイプラインとワークステーションから直接レンダリングプロジェクトとジョブを作成および管理できます。

AWSSDK、AWS Command Line Interface(AWS CLI)、または Deadline Cloud ジョブ送信者を使用して Deadline Cloud にジョブを送信します。Deadline Cloud は、ジョブテンプレート仕様の Open Job Description (OpenJD) をサポートしています。詳細については、GitHubウェブサイトの「[ジョブの説明を開く](#)」を参照してください。

Deadline Cloud はジョブ送信者を提供します。ジョブ送信者は、Mayaやなどのサードパーティーの DCC インターフェイスからレンダリングジョブを送信するための DCC プラグインですNuke。送信者を使用すると、アーティストはサードパーティーのインターフェイスから Deadline Cloud にレンダリングジョブを送信し、プロジェクトリソースが管理され、ジョブがモニタリングされます。

Deadline Cloud ファームを使用すると、キューとフリートの作成、ユーザーの管理、プロジェクトリソースの使用状況とコストの管理を行うことができます。ファームはキューとフリートで構成さ

れます。キューは、送信されたジョブが配置され、レンダリングがスケジュールされる場所です。フリートは、タスクを実行してジョブを完了するワーカーノードのグループです。ジョブをレンダリングするには、キューをフリートに関連付ける必要があります。1つのフリートで複数のキューをサポートでき、1つのキューを複数のフリートでサポートできます。

ジョブはステップで構成され、各ステップは特定のタスクで構成されます。Deadline Cloud モニターを使用すると、ジョブ、ステップ、タスクのステータス、ログ、その他のトラブルシューティングメトリクスにアクセスできます。

Deadline Cloud のアクセス許可

Deadline Cloud は以下をサポートしています。

- AWS Identity and Access Management(IAM) を使用した API オペレーションへのアクセスの管理
- との統合を使用したワークフォースユーザーのアクセスの管理AWS IAM アイデンティティセンター

誰でもプロジェクトに取り組む前に、そのプロジェクトと関連するファームにアクセスできる必要があります。Deadline Cloud は IAM Identity Center と統合され、ワークフォースの認証と認可を管理します。ユーザーは IAM Identity Center に直接追加することも、アクセス許可を Okta や などの既存の ID プロバイダー (IdP) に接続することもできますActive Directory。IT 管理者は、さまざまなレベルでユーザーとグループにアクセス許可を付与できます。後続の各レベルには、前のレベルのアクセス許可が含まれます。次のリストでは、最低レベルから最高レベルまでの 4 つのアクセスレベルについて説明します。

- ビューワー – アクセスできるファーム、キュー、フリート、ジョブ内のリソースを表示するアクセス許可。ビューワーはジョブを送信または変更できません。
- コントリビューター – ビューワーと同じですが、キューまたはファームにジョブを送信するアクセス許可があります。
- マネージャー – 寄稿者と同じですが、アクセスできるキュー内のジョブを編集し、アクセスできるリソースに対するアクセス許可を付与するアクセス許可があります。
- 所有者 – マネージャーと同じですが、予算を表示および作成し、使用状況を確認できます。

Note

これらのアクセス許可は、AWS マネジメントコンソールまたは Deadline Cloud インフラストラクチャを変更するアクセス許可をユーザーに付与しません。

ユーザーは、関連するキューとフリートにアクセスする前に、ファームにアクセスできる必要があります。ユーザーアクセスは、ファーム内でキューとフリートに個別に割り当てられます。

ユーザーを個人として、またはグループの一部として追加できます。ファーム、フリート、またはキューにグループを追加すると、大規模なグループのアクセス許可の管理が容易になります。たとえば、特定のプロジェクトに取り組んでいるチームがある場合は、各チームメンバーをグループに追加できます。次に、対応するファーム、フリート、またはキューのグループ全体にアクセス許可を付与できます。

Deadline Cloud でのソフトウェアサポート

Deadline Cloud は、コマンドラインインターフェイスから実行でき、パラメータ値を使用して制御できるソフトウェアアプリケーションと連携します。Deadline Cloud は、タスクにパラメータ化されたソフトウェアスクリプトステップ (フレーム範囲など) を使用して、作業をジョブとして記述するための OpenJD 仕様をサポートしています。Deadline Cloud ツールと機能を使用して OpenJD ジョブバンドルにジョブ指示をアSEMBルし、サードパーティーのソフトウェアアプリケーションからステップを作成、実行、ライセンスします。

ジョブをレンダリングするにはライセンスが必要です。Deadline Cloud は usage-based-licensing (UBL) を提供します。Deadline Cloud では、必要に応じて独自のソフトウェアライセンスを使用することもできます。ジョブがライセンスにアクセスできない場合、レンダリングされず、Deadline Cloud モニターのタスクログに表示されるエラーが生成されます。

Deadline Cloud をパイプラインに統合する

既存のレンダリングパイプラインを AWS Deadline Cloud と統合して、ワークフロー管理とジョブ送信プロセスを合理化できます。

パイプライン統合とは

Deadline Cloud のパイプライン統合とは、Deadline Cloud ファームがインタラクティブワークフローと自動ワークフローにバッチ処理を提供する方法を指します。この例では、オペレータがワーク

フローで使用するアプリケーションやプロセスに適応できるビジュアルエフェクトパイプラインを使用しています。

ビジュアルエフェクトパイプラインは、入力映像、3D モデル、アニメーション、テクスチャ、ライティング、レンダリングされた画像などを処理するポストプロダクションのステージで構成されます。さまざまな部門が、担当するタスクを実行するためにアセットを交換する方法を規定しています。適切に設計されたパイプラインにより、テレビ番組などの最終イメージを効率的に作成できます。

Deadline Cloud ファームをパイプラインに統合することで、長時間実行されるジョブをキューにオフロードし、Deadline Cloud がワーカーホストのフリートでジョブをスケジュールする方法を優先できます。サービスによって管理されるフリートを使用し、オンプレミスまたは独自のフリートを作成できます AWS。

パイプライン統合を作成するには、次の要素を考慮してください。

- アセットデータはどこに保存され、ファーム内のワーカーホストにどのように提供されますか？
- ジョブにはどのアプリケーションとプラグインが必要で、ファームのワーカーホストにプロビジョニングするにはどうすればよいですか？
- アーティストや他のオペレーターが実行するジョブがある場合、どのようにファームに送信しますか？
- 誰がジョブの進行状況とステータスをモニタリングし、どのようにコストを制御し、ワーカーホストの使用率を最適化しますか？

ファームが のオンプレミススタジオの例 AWS

この例では、アーティストがオンプレミスで作業し、レンダリング AWS のために のファームにジョブを送信するパイプラインに焦点を当てています。ここで紹介するアプローチは、Deadline Cloud にすばやくオンボードでき、カスタマイズのための柔軟な開始点を提供します。

このサンプルスタジオのパイプライン統合の要因は次のとおりです。

- アセットデータは、オンプレミスオフィスの NAS 共有ファイルシステムに保存されます。
 - ではWindows、プロジェクトは P: ドライブにマウントされ、ユーティリティは X: にマウントされます。
 - ではmacOS、プロジェクトは /Volumes/Projects にマウントされ、ユーティリティは /Volumes/Utilities にマウントされます。

- 3D モデリングには Maya、レンダリングには Arnold、合成には Nuke を使用します。これらのアプリケーションにはカスタムプラグインがインストールされていません。
- デフォルトの送信エクスペリエンスを使用します。
- アーティストは自分のジョブをモニタリングし、プロデューサーは必要に応じてコストをモニタリングし、優先順位を調整します。

このスタジオのパイプライン統合では、ジョブアタッチメントを使用して、スタジオの施設との間でデータを転送します。これは、の使用を簡単に開始でき AWS、大規模なフリートサイズにスケールできるためです。キューに設定されたジョブアタッチメント S3 バケットは、オンプレミス NAS とワーカーホスト間のキャッシュ階層として機能します AWS。

アーティストが Maya または Nuke からジョブを送信すると、Deadline Cloud 統合送信者はシーンをスキャンしてジョブの実行に必要なファイルを特定し、それらを S3 にアップロードしてジョブにアタッチします。高性能ハッシュは、スタジオ内のアーティストによって以前にアップロードされたファイルを識別するために使用されます。これにより、アーティストが同じショットの新しいバージョンを繰り返し送信している場合、またはあるアーティストが別のアーティストにショットを渡す場合、ジョブの送信プロセスでアップロードする必要があるのは、新規または変更されたファイルのみです。

スタジオは Windows と の両方の macOS ワークステーションを使用するため、プロジェクトとユーザーリテイドライブの両方のローカルタイプのファイルシステムの場所を使用してストレージプロファイルを設定します。ジョブの送信元とは異なるオペレーティングシステムでジョブが実行されたときに必要なパスマッピングをサポートする方法の詳細については、[ジョブアタッチメントのストレージプロファイル](#)のトピックを参照してください。また、完了時にキュー内のすべてのジョブタスクの出力を自動的にダウンロードするように、ネットワーク上の Linux ホストを設定します。設定方法については、[「ジョブアタッチメントの自動ダウンロード」](#)を参照してください。

ファームには、スタジオがジョブに必要な最小仕様から始まる範囲に設定された vCPUs と RAM 要件を持つ 2 つの Linux サービスマネージドフリートが含まれています。フリートの 1 つは、勤務時間中に一貫したレンダリング容量を提供するために少数のスポットインスタンスを提供するように設定され、もう 1 つのフリートは待機および保存として設定され、オフピーク時間により多くのジョブを低コストでレンダリングします。Maya、Maya for Arnold プラグイン、Nuke はすべて、使用ベースのライセンスとともに、期限クラウド conda チャンネルから Linux サービスマネージドフリート向けに提供されています。アプリケーションのインストールによるオーバーヘッドを節約するために、Deadline Cloud コンソールのキュー用に設定されたデフォルトの conda 環境を、[キャッシュが改善された github サンプル conda キュー環境に置き換えます](#)。

ジョブの送信をサポートするために、各ワークステーションに [Deadline Cloud 送信者を設定](#)し、Maya と Nuke の統合を選択します。Deadline Cloud モニターを使用すると、ファームにログインし、ジョブの進行状況をモニタリングし、問題を診断するためのログ出力を表示できます。Maya と Nuke の両方の送信元には、アプリケーションインターフェイス内からジョブを送信するための統合ダイアログがあります。

ファームで [ユーザーアクセスレベルを設定する](#) と、コントリビューターにアーティストへのアクセス権が付与されるため、コントリビューターはジョブの送信、すべてのジョブの表示、独自のジョブのプロパティの変更を行うことができます。これにより、マネージャーはすべてのジョブのプロパティを変更できるように、ラングラーをレンダリングするアクセス権が付与されます。予算を作成し、使用コストを調べることで、[支出と使用状況を追跡](#)できるように、所有者にプロデューサーへのアクセス権を付与します。

Deadline Cloud の開始方法

AWS Deadline Cloud でファームを作成するには、[Deadline Cloud コンソール](#)または AWS Command Line Interface () を使用できますAWS CLI。コンソールを使用して、キューやフリートなど、ファームの作成に関するガイド付きエクスペリエンスを提供します。を使用して AWS CLI、サービスを直接操作するか、Deadline Cloud で動作する独自のツールを開発します。

ファームを作成し、Deadline Cloud モニターを使用するには、Deadline Cloud のアカウントを設定します。Deadline Cloud モニターインフラストラクチャは、アカウントごとに 1 回だけセットアップする必要があります。ファームから、ファームとそのリソースへのユーザーアクセスを含むプロジェクトを管理できます。

ジョブを受け入れる最小限のリソースでファームを作成するには、コンソールのホームページでクイックスタートを選択します。では、これらの手順[Deadline Cloud モニターをセットアップする](#)について説明します。これらのファームは、キューと自動的に関連付けられるフリートで始まります。このアプローチは、実験するサンドボックススタイルのファームを作成するのに便利な方法です。

トピック

- [をセットアップする AWS アカウント](#)
- [Deadline Cloud モニターをセットアップする](#)
- [ワークステーションのセットアップ](#)

をセットアップする AWS アカウント

にサインアップする AWS アカウント

の使用を開始するには AWS、が必要です AWS アカウント。の作成の詳細については AWS アカウント、AWS アカウント管理 リファレンスガイドの「[の開始方法 AWS アカウント](#)」を参照してください。

Deadline Cloud モニターをセットアップする

開始するには、モニター、キュー、フリートなどの Deadline Cloud ファームインフラストラクチャを作成する必要があります。グループとユーザーの追加、サービスロールの選択、リソースへのタグの追加など、追加のオプション手順を実行することもできます。

ステップ 1: モニターを作成する

Deadline Cloud モニターは を使用してユーザーを承認 AWS IAM アイデンティティセンターします。デフォルトでは、Deadline Cloud に使用する IAM Identity Center インスタンスは、モニター AWS リージョンと同じ がある必要があります。ただし、IAM Identity Center でマルチリージョンサポートが有効になっている場合は、別のリージョンにモニターを作成できます。詳細については、「[AWS IAM アイデンティティセンターとは](#)」を参照してください。モニターの作成時にコンソールで別のリージョンを使用している場合は、IAM Identity Center リージョンへの変更に関するリマインダーが表示されます。

モニターのインフラストラクチャは、次のコンポーネントで構成されます。

- **モニター名:** Monitor 名は、AnyCompany モニターなど、モニターを識別する方法です。モニターの名前によって、モニター URL も決まります。
- **モニター URL:** モニター URL を使用してモニターにアクセスできます。URL はモニター名に基づいています。例: `https://anycompanymonitor.awsapps.com`。
- **AWS リージョン:** AWS リージョンは、AWS データセンターの集合体の物理的な場所です。モニターを設定すると、リージョンはデフォルトで最も近い場所に設定されます。ユーザーに最も近いリージョンに変更することをお勧めします。これにより遅延が減少し、データ転送速度が向上します。デフォルトでは、IAM アイデンティティセンターでマルチリージョンサポートを有効にしている限り、 は Deadline Cloud AWS リージョンと同じ で有効に AWS IAM アイデンティティセンター する必要があります。詳細については、「[とは AWS IAM アイデンティティセンター](#)」を参照してください。

Important

Deadline Cloud の設定が完了したら、リージョンを変更することはできません。

このセクションのタスクを完了して、モニターのインフラストラクチャを設定します。

モニターのインフラストラクチャを設定するには

1. にサインインAWS マネジメントコンソールして、Welcome to Deadline Cloud のセットアップを開始し、次へを選択します。
2. Monitor 名を入力します。例: **AnyCompany Monitor**。
3. (オプション) Monitor URL を変更するには、URL の編集を選択します。

4. (オプション) ユーザーに最も近いAWS リージョンのように を変更するには、リージョンの変更を選択します。
 - a. ユーザーに最も近いリージョンを選択します。
 - b. [リージョンを適用] を選択します。
5. (オプション) モニターの設定をさらにカスタマイズするには、 を選択します [詳細設定](#)。
6. の準備ができたなら [ステップ 2: ファームの詳細を定義する](#)、次へを選択します。

詳細設定

Deadline Cloud のセットアップには、追加の設定が含まれます。これらの設定を使用すると、Deadline Cloud のセットアップによって に加えられたすべての変更を表示したり AWS アカウント、モニターユーザーロールを設定したり、暗号化キータイプを変更したりできます。

AWS IAM アイデンティティセンター

AWS IAM アイデンティティセンター は、ユーザーとグループを管理するためのクラウドベースのシングルサインオンサービスです。IAM Identity Center をエンタープライズシングルサインオン (SSO) プロバイダーと統合して、ユーザーが会社のアカウントでサインインできるようにすることも可能です。

Deadline Cloud はデフォルトで IAM Identity Center を有効にし、Deadline Cloud をセットアップして使用する必要があります。デフォルトでは、Deadline Cloud に使用する IAM Identity Center インスタンスは、モニター AWS リージョン と同じ 必要があります。ただし、IAM Identity Center でマルチリージョンサポートが有効になっている場合は、別のリージョンにモニターを作成できます。詳細については、[「とは AWS IAM アイデンティティセンター」](#) を参照してください。

サービスアクセスロールを設定する

AWS サービスは、ユーザーに代わってアクションを実行するサービスロールを引き受けることができます。Deadline Cloud では、モニター内のリソースへのアクセス権をユーザーに付与するために、モニターユーザーロールが必要です。

AWS Identity and Access Management (IAM) 管理ポリシーをモニターユーザーロールにアタッチできます。このポリシーは、特定の Deadline Cloud アプリケーションでジョブを作成するなど、特定のアクションを実行するアクセス許可をユーザーに付与します。アプリケーションは管理ポリシーの特定の条件に依存するため、管理ポリシーを使用しないと、アプリケーションが期待どおりに動作しない可能性があります。

モニターユーザーロールは、セットアップの完了後にいつでも変更できます。ユーザーロールの詳細については、「[IAM ロール](#)」を参照してください。

以下のタブには、2つの異なるユースケースの説明が含まれています。新しいサービスロールを作成して使用するには、[新しいサービスロール] タブを選択します。既存のサービスロールを使用するには、[既存のサービスロール] タブを選択します。

New service role

新しいサービスロールを作成して使用するには

1. [新しいサービスロールを作成し使用する] を選択します。
2. (オプション) サービスユーザーロール名を入力します。
3. ロールの詳細については、[許可の詳細を表示] を選択します。

Existing service role

既存のサービスロールを使用するには

1. [既存のサービスロールを使用する] を選択します。
2. ドロップダウンリストを開いて既存のサービスロールを選択します。
3. (オプション) ロールの詳細については、IAM コンソールで表示を選択します。

ステップ 2: ファームの詳細を定義する

Deadline Cloud コンソールに戻り、次のステップを実行してファームの詳細を定義します。

1. Farm の詳細で、ファームの名前を追加します。
2. 説明 にファームの説明を入力します。説明は、ファームの目的を特定するのに役立ちます。
3. グループを作成し、ファームの用途を追加します。ファームを設定したら、Deadline Cloud マネジメントコンソールを使用してグループとユーザーを追加または変更できます。
4. (オプション) 追加のファーム設定を選択します。
 - a. (オプション) デフォルトでは、データはセキュリティのために が AWS 所有および管理するキーで暗号化されます。暗号化設定をカスタマイズ (詳細) を選択して、既存のキーを使用するか、管理する新しいキーを作成できます。

チェックボックスを使用して暗号化設定をカスタマイズする場合は、ARN AWS KMS を入力するか、新しい KMS キーの作成 AWS KMS を選択して新しい を作成します。

- b. (オプション) 新しいタグを追加を選択して、ファームに 1 つ以上のタグを追加します。
5. 以下のオプションのいずれかを選択してください。
- 「スキップして確認」と「作成」を選択し、[ファームを確認して作成します](#)。
 - 次へ を選択して、追加のオプションステップに進みます。

(オプション) ステップ 3: キューの詳細を定義する

キューは、ジョブの進行状況を追跡し、作業をスケジュールします。

1. キューの詳細から、キューの名前を指定します。
2. 説明 にキューの説明を入力します。明確な説明は、キューの目的をすばやく特定するのに役立ちます。
3. ジョブアタッチメントでは、新しい Amazon S3 バケットを作成するか、既存の Amazon S3 バケットを選択できます。既存の Amazon S3 バケットがない場合は、バケットを作成する必要があります。
 - a. 新しい Amazon S3 バケットを作成するには、新しいジョブバケットの作成を選択します。ルートプレフィックスフィールドでジョブバケットの名前を定義できます。バケットを呼び出すことをお勧めします `deadlinecloud-job-attachments-[QUEUENAME]`。

小文字とダッシュのみを使用できます。スペースや特殊文字は使用できません。
 - b. 既存の Amazon S3 バケットを検索して選択するには、既存の Amazon S3 バケットから選択を選択します。次に、Browse S3 を選択して既存のバケットを検索します。使用可能な Amazon S3 バケットのリストが表示されたら、キューに使用する Amazon S3 バケットを選択します。
4. (オプション) 追加のファーム設定を選択します。
 - a. カスタマーマネージドフリートを使用している場合は、カスタマーマネージドフリートとの関連付けを有効にするを選択します。
 - i. カスタマーマネージドフリートの場合は、キュー設定ユーザーを追加し、POSIX および/または Windows 認証情報を設定します。または、チェックボックスを選択して `run-as` 機能をバイパスすることもできます。

- ii. キューの予算を設定する場合は、このキューの予算を要求するを選択します。予算が必要な場合は、Deadline Cloud コンソールを使用して予算を作成し、キュー内のジョブをスケジュールする必要があります。
- b. キューには、ユーザーに代わって Amazon S3 にアクセスするためのアクセス許可が必要です。キューごとに新しいサービスロールを作成することをお勧めします。
 - i. 新しいロールの場合は、次の手順を実行します。
 - A. [新しいサービスロールを作成し使用する] を選択します。
 - B. キューロールのロール名を入力するか、指定されたロール名を使用します。
 - C. (オプション) キューロールの説明を追加します。
 - D. アクセス許可の詳細を表示を選択して、キューロールの IAM アクセス許可を表示できます。
 - ii. または、既存のサービスロールを選択することもできます。
- c. (オプション) 名前と値のペアを使用して、キュー環境の環境変数を追加します。
- d. (オプション) キーと値のペアを使用してキューにタグを追加します。

以下のオプションのいずれかを選択してください。

- 「スキップして確認」と「作成」を選択し、[フォームを確認して作成します](#)。
- 次へ を選択して、追加のオプションステップに進みます。

(オプション) ステップ 4: フリートの詳細を定義する

フリートは、レンダリングタスクを実行するワーカーを割り当てます。レンダリングタスクにフリートが必要な場合は、フリートの作成のチェックボックスをオンにします。

1. フリートの詳細
 - a. フリートの名前とオプションの説明の両方を指定します。
 - b. フリートタイプとオペレーティングシステムの認識を確認します。
2. インスタンス市場タイプセクションで、スポット、オンデマンド、またはインスタンスの待機と保存を選択します。Amazon EC2 オンデマンドインスタンスはより高速な可用性を提供し、Amazon EC2 スポットインスタンスと Wait and Save インスタンスはコスト削減の取り組みに適しています。

3. フリート内のインスタンス数を自動スケーリングするには、最小インスタンス数と最大インスタンス数の両方を選択します。

追加コストが発生しないように、常にインスタンスの最小数0を に設定することを強くお勧めします。
4. ワーカーの認識度を確認します。
5. (オプション) 追加のフリート設定を選択する
 - a. フリートには、ユーザーに代わって CloudWatch に書き込むためのアクセス許可が必要です。フリートごとに新しいサービスロールを作成することをお勧めします。
 - i. 新しいロールの場合は、次の手順を実行します。
 - A. [新しいサービスロールを作成し使用する] を選択します。
 - B. フリートロールのロール名を入力するか、指定されたロール名を使用します。
 - C. (オプション) フリートロールの説明を追加します。
 - D. フリートロールの IAM アクセス許可を表示するには、アクセス許可の詳細を表示するを選択します。
 - ii. または、既存のサービスロールを使用することもできます。
 - b. (オプション) キーと値のペアを使用してフリートのタグを追加します。

すべてのフリートの詳細を入力したら、次へを選択します。

ステップ 5: 確認して作成する

入力した情報を確認してファームを作成します。準備ができたら、ファームの作成を選択します。

ファームの作成の進行状況が Farms ページに表示されます。ファームを使用する準備が整うと、成功メッセージが表示されます。

ワークステーションのセットアップ

このプロセスは、AWS Deadline Cloud 送信者をインストール、セットアップ、起動したい管理者とアーティストを対象としています。Deadline Cloud 送信者は、デジタルコンテンツ作成 (DCC) プラグインです。アーティストはこれを使用して、使い慣れたサードパーティーの DCC インターフェイスからジョブを送信します。

Note

このプロセスは、アーティストがレンダリングの送信に使用するすべてのワークステーションで完了する必要があります。

各ワークステーションには、対応する送信者をインストールする前に DCC がインストールされている必要があります。たとえば、の Deadline Cloud 送信者をダウンロードする場合はBlender、ワークステーションに がBlender既にインストールされている必要があります。

ワークステーションを安全に保つための合理的なデフォルトが用意されています。ワークステーションの保護の詳細については、[「セキュリティのベストプラクティス - ワークステーション」](#)を参照してください。

トピック

- [ステップ 1: Deadline Cloud 送信者をインストールする](#)
- [ステップ 2: Deadline Cloud モニターをインストールしてセットアップする](#)
- [ステップ 3: Deadline Cloud 送信者を起動する](#)

ステップ 1: Deadline Cloud 送信者をインストールする

以下のセクションでは、Deadline Cloud 送信者をインストールする手順について説明します。

Note

Unreal Engine: Unreal Engine 送信者は標準インストーラに含まれていないため、別のセットアッププロセスが必要です。インストール手順については、[Unreal Engine Submitter Setup Guide](#)を参照してください。

送信者インストーラーをダウンロードする

Deadline Cloud 送信者をインストールする前に、送信者インストーラーをダウンロードする必要があります。

1. オペレーティングシステムの送信者インストーラーをダウンロードします。

[Windows 用のダウンロード](#)[Linux 用のダウンロード](#)[MacOS 用ダウンロード \(arm64\)](#)

2. (オプション) [ダウンロードしたソフトウェアの信頼性を検証する](#)。

Deadline Cloud 送信者をインストールする

インストーラでは、次の送信者をインストールできます。

ソフトウェア	サポートバージョン	Windows インストーラ	Linux インストーラ	MacOS (arm64) インストーラ
Adobe After Effects	2024 ~ 2026 年	含まれる	含まれない	含まれる
Autodesk 3ds Max	2024 ~ 2026 年	含まれる	含まれない	含まれない
Autodesk Arnold for Cinema 4D	4.8.4.1	含まれる	含まれない	含まれる
Autodesk Arnold for Maya	7.1 ~ 7.4	含まれる	含まれる	含まれる
Autodesk Maya	2023 ~ 2026 年	含まれる	含まれる	含まれる
Autodesk VRED	2025 ~ 2026 年	含まれる	含まれない	含まれない
ブレンダー	3.6 ~ 5.1	含まれる	含まれる	含まれる
Chaos V-Ray for Maya	6 ~ 7	含まれる	含まれる	含まれる
Foundry Nuke	15 ~ 17	含まれる	含まれる	含まれる
KeyShot Studio	2023 ~ 2025 年	含まれる	含まれない	含まれる

ソフトウェア	サポートバージョン	Windows インストーラ	Linux インストーラ	MacOS (arm64) インストーラ
Maxon シネマ 4D	2024 ~ 2026 年	含まれる	含まれない	含まれる
Maxon Redshift for Maya	2025-2026	含まれる	含まれる	含まれる
SideFX フーディーニ	19.5 ~ 21.0	含まれる	含まれる	含まれる

Note

Unreal Engine: Unreal Engine 送信者は標準インストーラに含まれていないため、別のセットアッププロセスが必要です。インストール手順については、[Unreal Engine Submitter Setup Guide](#) を参照してください。

Windows

- ファイルブラウザで、インストーラがダウンロードしたフォルダに移動し、 を選択します。DeadlineCloudSubmitter-windows-x64-installer.exe。
 - Windows で保護されている PC ポップアップが表示された場合は、詳細を選択します。
 - とにかく実行を選択します。
- Deadline Cloud Submitter Setup Wizard AWS が開いたら、次へを選択します。
- 次のいずれかのステップを実行して、インストールスコープを選択します。
 - 現在のユーザーのみに をインストールするには、ユーザーを選択します。
 - すべてのユーザーに をインストールするには、システムを選択します。

System を選択した場合は、インストーラを終了し、次の手順を実行して管理者として再実行する必要があります。

- を右クリックし **DeadlineCloudSubmitter-windows-x64-installer.exe**、管理者として実行を選択します。
- 管理者認証情報を入力し、「はい」を選択します。

- c. インストールスコープのシステムを選択します。
4. インストールスコープを選択したら、次へを選択します。
5. インストールディレクトリを受け入れるには、もう一度次へを選択します。
6. の統合送信者Nuke、またはインストールする送信者を選択します。
7. [次へ] を選択します。
8. インストールを確認し、次へを選択します。
9. 「次へ」をもう一度選択し、「完了」を選択します。

Linux

Note

Linux および Deadline Cloud Monitor 用の Deadline Cloud 統合Nukeインストーラは、少なくとも GLIBC 2.31 のLinuxディストリビューションにのみインストールできます。

1. ターミナルウィンドウを開きます。
2. インストーラのシステムインストールを実行するには、コマンドを入力し **sudo -i**、Enter キーを押してルートにします。
3. インストーラをダウンロードした場所に移動します。

例えば、**cd /home/*USER*/Downloads**。

4. インストーラを実行可能にするには、「**」**と入力します **chmod +x DeadlineCloudSubmitter-linux-x64-installer.run**。
5. Deadline Cloud 送信者インストーラを実行するには、**./DeadlineCloudSubmitter-linux-x64-installer.run** と入力します。
6. インストーラが開いたら、画面のプロンプトに従ってセットアップウィザードを完了します。

macOS (arm64)

1. ファイルブラウザで、インストーラがダウンロードしたフォルダに移動し、ファイルを選択します。
2. Deadline Cloud Submitter Setup Wizard AWS が開いたら、次へを選択します。

3. インストールディレクトリを受け入れるには、もう一度次へを選択します。
4. の統合送信者Maya、またはインストールする送信者を選択します。
5. [次へ] を選択します。
6. インストールを確認し、次へを選択します。
7. 「次へ」をもう一度選択し、「終了」を選択します。

ステップ 2: Deadline Cloud モニターをインストールしてセットアップする

Deadline Cloud Monitor デスクトップアプリケーションは、Windows、Linuxまたは macOS を使用してインストールできます。

Windows

1. の Deadline Cloud Monitor インストーラをダウンロードしますWindows。

[Windows 用の Deadline Cloud モニターをダウンロードする](#)

2. ダウンロードしたインストーラを実行し、プロンプトに従ってインストールを完了します。

サイレントインストールを実行するには、次のコマンドを使用します。

```
DeadlineCloudMonitor_x64-setup.exe /S
```

デフォルトでは、モニターは にインストールされますC:\Users{username}\AppData\Local\DeadlineCloudMonitor。インストールディレクトリを変更するには、代わりに次のコマンドを使用します。

```
DeadlineCloudMonitor_x64-setup.exe /S /D={InstallDirectory}
```

Linux (Applmage)

Debian ディストリビューションに Deadline Cloud Monitor Applmage をインストールするには

1. Deadline Cloud Monitor Applmage をダウンロードします。

[Deadline Cloud Monitor のダウンロード \(Applmage\)](#)

2.

Note

このステップは Ubuntu 22 以降を対象としています。Ubuntu の他のバージョンでは、このステップをスキップします。

libfuse2 をインストールするには、次のように入力します。

```
sudo apt update
sudo apt install libfuse2
```

3. AppImage を実行可能にするには、次のように入力します。

```
chmod a+x deadline-cloud-monitor_amd64.AppImage
```

Linux (Debian)

Debian ディストリビューションに Deadline Cloud Monitor Debian パッケージをインストールするには

1. Deadline Cloud Monitor Debian パッケージをダウンロードします。

[Deadline Cloud Monitor のダウンロード \(.deb\)](#)

2.

Note

このステップは Ubuntu 22 以降用です。Ubuntu の他のバージョンでは、このステップをスキップします。

libssl1.1 をインストールするには、次のように入力します。

```
wget https://archive.ubuntu.com/ubuntu/pool/main/o/openssl/
libssl1.1_1.1.1f-1ubuntu2_amd64.deb
sudo apt install ./libssl1.1_1.1.1f-1ubuntu2_amd64.deb
```

3. Deadline Cloud Monitor Debian パッケージをインストールするには、次のように入力します。

```
sudo apt update
sudo apt install ./deadline-cloud-monitor_amd64.deb
```

4. 依存関係が満たされていないパッケージでインストールが失敗した場合、壊れたパッケージを修正し、次のコマンドを実行します。

```
sudo apt --fix-missing update
sudo apt update
sudo apt install -f
```

Linux (RPM)

Rocky Linux 9 または に Deadline Cloud Monitor RPM をインストールするには Alma Linux 9

Note

Rocky Linux 9 とはデフォルトで OpenSSL 3.0 Alma Linux 9を使用し、libssl.so.1.1ライブラリは含まれません。Deadline Cloud モニターを実行するには、compat-openssl11パッケージをインストールする必要があります。

1. Deadline Cloud Monitor RPM をダウンロードします。

[Deadline Cloud Monitor \(.rpm\) をダウンロードする](#)


2. Enterprise Linux 9 リポジトリの追加のパッケージを追加します。

```
sudo dnf install epel-release
```

3. libssl.so.1.1 依存関係compat-openssl11に をインストールします。

```
sudo dnf install compat-openssl11 deadline-cloud-monitor.x86_64.rpm
```

に Deadline Cloud Monitor RPM をインストールするには Red Hat Linux 9

 Note

Red Hat Linux 9 はデフォルトで OpenSSL 3.0 を使用し、libssl.so.1.1ライブラリは含まれません。Deadline Cloud モニターを実行するには、compat-openssl11パッケージをインストールする必要があります。

1. Deadline Cloud Monitor RPM をダウンロードします。

[Deadline Cloud Monitor \(.rpm\) をダウンロードする](#)

2. CodeReady Linux Builder リポジトリを有効にします。

```
subscription-manager repos --enable codeready-builder-for-rhel-9-x86_64-rpms
```

3. の追加パッケージをインストールしますEnterprise RPM。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

4. libssl.so.1.1 依存関係compat-openssl11に をインストールします。

```
sudo dnf install compat-openssl11 deadline-cloud-monitor.x86_64.rpm
```

Deadline Cloud Monitor RPM を Rocky Linux 8、Alma Linux 8、または にインストールするには Red Hat Linux 8

1. Deadline Cloud Monitor RPM をダウンロードします。

[Deadline Cloud Monitor のダウンロード \(.rpm\)](#)

2. Deadline Cloud モニターをインストールします。

```
sudo dnf install deadline-cloud-monitor.x86_64.rpm
```

macOS (arm64)

1. の Deadline Cloud Monitor インストーラをダウンロードしますmacOS。

[macOS \(arm64\) 用の Deadline Cloud モニターをダウンロードする](#)

- ダウンロードしたファイルを開きます。ウィンドウが表示されたら、Deadline Cloud モニターアイコンを選択してアプリケーションフォルダにドラッグします。

ダウンロードが完了したら、ダウンロードしたソフトウェアの信頼性を検証できます。ダウンロードプロセス中またはダウンロードプロセス後に、誰もファイルを改ざんしていないことを確認するために、これを行うことをお勧めします。ステップ 1 [ダウンロードしたソフトウェアの信頼性を検証する](#)の「」を参照してください。

Deadline Cloud モニターをダウンロードして信頼性を確認したら、次の手順を使用して Deadline Cloud モニターを設定します。

Deadline Cloud モニターを設定するには

- Deadline Cloud Monitor を開きます。
- 新しいプロファイルを作成するように求められたら、次の手順を実行します。
 - モニター URL を URL 入力に入力すると、次のようになります。 **`https://MY-MONITOR.deadlinecloud.amazonaws.com/`**
 - プロファイル名を入力します。
 - プロファイルの作成 を選択します。

プロファイルが作成され、作成したプロファイル名を使用するソフトウェアと認証情報が共有されるようになりました。

- Deadline Cloud モニタープロファイルを作成した後、プロファイル名またはスタジオ URL を変更することはできません。変更する必要がある場合は、代わりに次の操作を行います。
 - プロファイルを削除します。左側のナビゲーションペインで、Deadline Cloud monitor > Settings > Delete を選択します。
 - 必要な変更を含む新しいプロファイルを作成します。
- 左側のナビゲーションペインで、>Deadline Cloud monitor オプションを使用して以下を実行します。
 - Deadline Cloud モニタープロファイルを変更して、別のモニターにログインします。
 - 自動ログインを有効にすると、以降の Deadline Cloud Monitor のオープン時にモニター URL を入力する必要がなくなります。

5. Deadline Cloud モニターウィンドウを閉じます。バックグラウンドで実行され続け、他の Deadline Cloud ツールがレンダーファームにアクセスできるようにします。
6. レンダリングプロジェクトに使用する予定のデジタルコンテンツ作成 (DCC) アプリケーションごとに、次の手順を実行します。
 - a. Deadline Cloud 送信者から、Deadline Cloud ワークステーション設定を開きます。
 - b. ワークステーション設定で、Deadline Cloud モニターで作成したプロファイルを選択します。Deadline Cloud 認証情報がこの DCC と共有され、ツールは期待どおりに動作するはずです。

ステップ 3: Deadline Cloud 送信者を起動する

次の例は、Blender 送信者をインストールする方法を示しています。同様のステップを使用して、他の送信者をインストールできます。

で Deadline Cloud 送信者を起動するには Blender

Note

のサポートBlenderは、サービスマネージドフリートの conda環境を使用して提供されます。詳細については、「[デフォルトのcondaキュー環境](#)」を参照してください。

1. Blender を開きます。
2. レンダリングメニューで、AWS Deadline Cloud に送信を選択します。
 - a. Deadline Cloud 送信者でまだ認証されていない場合、認証情報ステータスは NEEDS_LOGIN と表示されます。
 - b. [ログイン] を選択します。ブラウザでユーザー認証情報を使用してログインするように求められます。
 - c. これでログインし、認証情報のステータスが AUTHENTICATED と表示されます。
3. [Submit] を選択してください。

これで、ジョブが Deadline Cloud ファームに送信され、互換性のあるフリートによって処理されます。モニターでジョブの進行状況を表示する方法については、「[モニターの使用](#)」を参照してください。

Deadline Cloud モニターの使用

AWS Deadline Cloud モニターには、ビジュアルコンピューティングジョブの全体像が表示されます。これを使用して、ジョブのモニタリングと管理、フリートでのワーカーアクティビティの表示、予算と使用状況の追跡、ジョブの結果のダウンロードを行うことができます。

各キューには、ジョブ、ステップ、タスクのステータスを示すジョブモニターがあります。モニターには、モニターから直接ジョブを管理する方法が用意されています。優先順位付けの変更、ジョブのキャンセル、ジョブの再キュー、ジョブの再送信を行うことができます。

Deadline Cloud モニターには、ジョブの概要ステータスを示すテーブルがあります。または、ジョブを選択して、ジョブに関する問題のトラブルシューティングに役立つ詳細なタスクログを表示できます。

Deadline Cloud モニターを使用して、ジョブの作成時に指定されたワークステーション上の場所に結果をダウンロードできます。

Deadline Cloud モニターは、使用状況のモニタリングやコストの管理にも役立ちます。詳細については、「[Deadline Cloud フォームの支出と使用状況を追跡する](#)」を参照してください。

トピック

- [追加のリージョンでモニターを作成する](#)
- [Deadline Cloud モニター URL を共有する](#)
- [Deadline Cloud モニターを開く](#)
- [ジョブバンドルを送信する](#)
- [Deadline Cloud でキューとフリートの詳細を表示する](#)
- [Deadline Cloud でジョブ、ステップ、タスクを管理する](#)
- [Deadline Cloud でのジョブの詳細の表示と管理](#)
- [Deadline Cloud でステップを表示する](#)
- [Deadline Cloud でタスクを表示する](#)
- [Deadline Cloud でセッションログとワーカーログを表示する](#)
- [ワーカーダッシュボードでワーカーの詳細を表示する](#)
- [Deadline Cloud で完成した出力をダウンロードする](#)

- [Deadline Cloud でのジョブアタッチメントの表示](#)
- [Deadline Cloud Monitor デスクトップのデプロイとワークフローを自動化する](#)

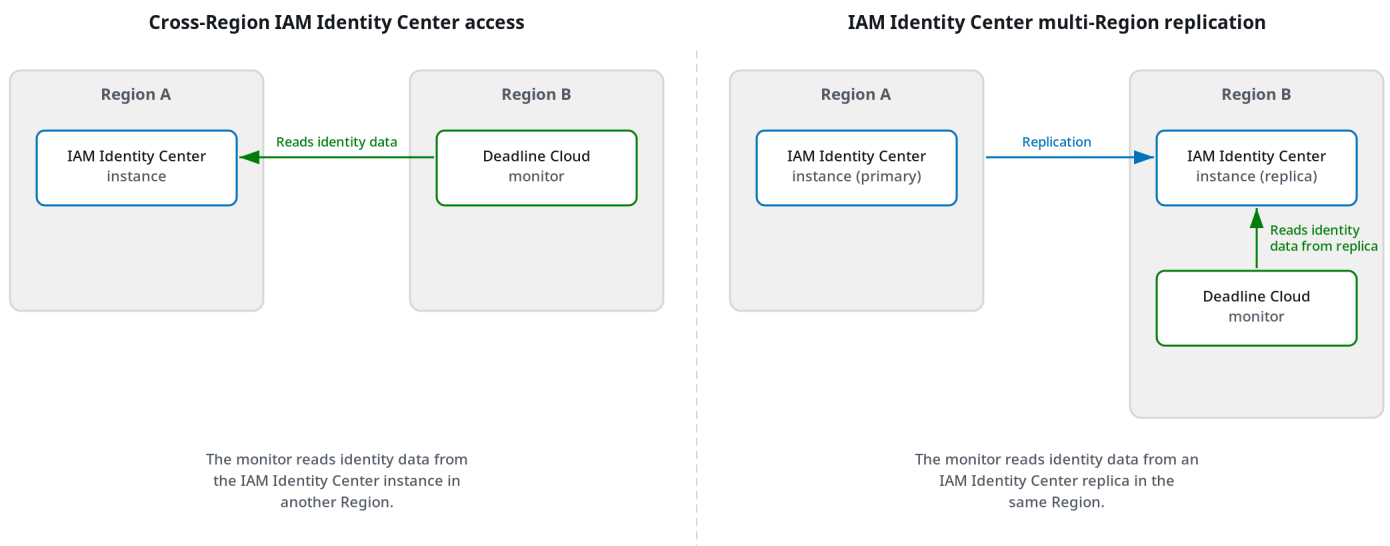
追加のリージョンでモニターを作成する

各 Deadline Cloud モニターは、リソースを 1 つの で管理します AWS リージョン。追加のリージョンでリソースを管理するには、リージョンごとに個別のモニターを作成できます。

モニターを作成するリージョンで IAM Identity Center インスタンスを使用できない場合は、次のオプションがあります。

- クロスリージョン IAM アイデンティティセンターへのアクセス – 別のリージョンにモニターを作成すると、Deadline Cloud は IAM アイデンティティセンターインスタンスが配置されているリージョンから IAM アイデンティティセンターのアイデンティティデータを読み取ります。このオプションでは、IAM アイデンティティセンターの設定を変更する必要はありません。
- IAM Identity Center マルチリージョンレプリケーション – IAM Identity Center インスタンスを追加のリージョンにレプリケートし、それらのリージョンのモニターが同じリージョンの IAM Identity Center レプリカを使用するようにします。このオプションはレイテンシーとリージョンの可用性を低くしますが、追加の IAM アイデンティティセンター設定が必要です。

次の図は、各アプローチの仕組みを示しています。



次の表は、2つのアプローチを比較したものです。

マルチリージョンアプローチの比較

考慮事項	クロスリージョン IAM アイデンティティセンターへのアクセス	IAM Identity Center マルチリージョンレプリケーション
セットアップ要件	追加の IAM アイデンティティセンターのセットアップは必要ありません	IAM Identity Center レプリケーションの設定が必要
ID データの場所	IAM Identity Center リージョンにのみ残ります	設定された各リージョンにレプリケート
レイテンシー	IAM Identity Center リージョンまでの距離によって異なります	IAM Identity Center レプリカが同じリージョンにある場合のレイテンシーの短縮
リージョン別の可用性	IAM Identity Center リージョンの可用性によって異なります	IAM アイデンティティセンターのプライマリリージョンが利用できない場合でも引き続き動作します

クロスリージョン IAM アイデンティティセンターへのアクセス

クロスリージョン IAM アイデンティティセンターアクセスでは、IAM アイデンティティセンターインスタンスとは異なるリージョンに Deadline Cloud モニターを作成します。Deadline Cloud は、IAM Identity Center インスタンスが配置されているリージョンから IAM Identity Center ID データを読み取ります。

Deadline Cloud コンソールを使用してモニターを作成すると、インスタンスが別のリージョンにある場合でも、コンソールは IAM Identity Center インスタンスを自動的に検出し、モニターを接続します。AWS SDK を使用してモニターを作成するときは、IAM Identity Center インスタンスがあるリージョンを指定します。

考慮事項

- クロスリージョン IAM アイデンティティセンターへのアクセスには、IAM アイデンティティセンターインスタンスが商用である必要があります AWS リージョン。オプトインリージョンの IAM Identity Center インスタンスはサポートされていません。

- モニターの作成後に IAM Identity Center リージョンを変更することはできません。

IAM Identity Center マルチリージョンレプリケーション

IAM アイデンティティセンターのマルチリージョンレプリケーションは、ユーザー、グループ、グループメンバーシップなどの IAM アイデンティティセンターのアイデンティティストアデータを追加の に同期します AWS リージョン。リージョンへのレプリケーションを有効にしたら、そのリージョンのモニターを IAM Identity Center レプリカに接続できます。

マルチリージョンレプリケーションは、以下のシナリオで役立ちます。

- レプリケートされたリージョンに近いユーザーの場合、レイテンシーを短くする必要があります。
- IAM アイデンティティセンターのプライマリリージョンが使用できない場合は、引き続き動作するモニターが必要です。

マルチリージョンレプリケーションを有効にするには、[「IAM Identity Center ユーザーガイド」の「Using IAM Identity Center across multiple AWS リージョン」](#)を参照してください。リージョンのレプリケーションを有効にしたら、コンソールまたは AWS SDK を使用して Deadline Cloud Monitor を作成できます。

Deadline Cloud モニター URL を共有する

Deadline Cloud サービスを設定すると、デフォルトでアカウントの Deadline Cloud モニターを開く URL が作成されます。この URL を使用して、ブラウザまたはデスクトップでモニターを開きます。Deadline Cloud モニターにアクセスできるように、他のユーザーと URL を共有します。

ユーザーが Deadline Cloud モニターを開く前に、ユーザーにアクセス権を付与する必要があります。アクセスを許可するには、モニターの承認されたユーザーのリストにユーザーを追加するか、モニターにアクセスできるグループに追加します。詳細については、[「Deadline Cloud でのユーザーの管理」](#)を参照してください。

モニター URL を共有するには

1. [Deadline Cloud コンソール](#)を開きます。
2. 開始するには、「Deadline Cloud ダッシュボードに移動」を選択します。
3. ナビゲーションペインで、ダッシュボードを選択します。
4. アカウントの概要セクションで、アカウントの詳細を選択します。

5. Deadline Cloud モニターにアクセスする必要があるすべてのユーザーに URL をコピーして安全に送信します。

Deadline Cloud モニターを開く

Deadline Cloud モニターは、次のいずれかの方法で開くことができます。

- コンソール – にサインイン AWS マネジメントコンソール し、Deadline Cloud コンソールを開きます。
- ウェブ – Deadline Cloud のセットアップ時に作成したモニター URL に移動します。
- Monitor – デスクトップの Deadline Cloud モニターを使用します。

コンソールを使用する場合、ID AWS を使用して AWS Identity and Access Management にサインインし、AWS IAM アイデンティティセンター 認証情報を使用してモニターにサインインする必要があります。IAM アイデンティティセンターの認証情報のみがある場合は、モニター URL またはデスクトップアプリケーションを使用してサインインする必要があります。

Deadline Cloud モニターを開くには (ウェブ)

1. ブラウザを使用して、Deadline Cloud のセットアップ時に作成したモニター URL を開きます。
2. ユーザー認証情報を使用してサインインします。

Deadline Cloud モニターを開くには (コンソール)

1. [Deadline Cloud コンソール](#)を開きます。
2. ナビゲーションペインで、ファームを選択します。
3. ファームを選択し、ジョブの管理を選択して Deadline Cloud モニターページを開きます。
4. ユーザー認証情報を使用してサインインします。

Deadline Cloud モニターを開くには (デスクトップ)

1. [Deadline Cloud コンソール](#)を開きます。

-または-

Deadline Cloud モニター - モニター URL からウェブを開きます。

2.
 - Deadline Cloud コンソールで、次の操作を行います。
 1. モニターで、Deadline Cloud ダッシュボードに移動を選択し、左側のメニューからダウンロードを選択します。
 2. Deadline Cloud モニターから、デスクトップのモニターバージョンを選択します。
 3. [ダウンロード] を選択します。
 - Deadline Cloud モニター - ウェブで、次の操作を行います。
 - 左側のメニューから、ワークステーションのセットアップを選択します。ワークステーションのセットアップ項目が表示されない場合は、矢印を使用して左側のメニューを開きます。
 - [ダウンロード] を選択します。
 - OS の選択 から、オペレーティングシステムを選択します。
3. Deadline Cloud Monitor - デスクトップをダウンロードします。
4. モニタをダウンロードしてインストールしたら、コンピュータで開きます。
 - Deadline Cloud モニターを初めて開く場合は、モニター URL を指定してプロファイル名を作成する必要があります。次に、Deadline Cloud 認証情報を使用してモニターにサインインします。
 - プロファイルを作成したら、プロファイルを選択してモニターを開きます。Deadline Cloud 認証情報の入力が必要になる場合があります。

言語設定を変更する

Deadline Cloud モニターを作成して開いたら、言語設定を変更できます。デフォルトでは、モニター言語はシステムの言語設定に設定されます。

Deadline Cloud Monitor (デスクトップ) から言語設定を変更するには

1. ユーザープロファイルから、設定 を選択し、言語 を選択します。
2. ドロップダウンメニューから、使用可能な言語のいずれかを選択します。
3. 選択した言語がリストされたオプションであることを確認し、確認を選択して変更を適用します。

モニターが更新されると、選択した言語で表示されます。

言語設定を変更した後は、 を開くとデフォルトになり、再度変更するかデスクトップアプリケーションをアンインストールするまでデフォルトのままになります。

ウェブで Deadline Cloud モニター言語を変更するには、ブラウザの設定で優先言語を変更します。

Note

ブラウザまたはオペレーティングシステムが Deadline Cloud でサポートされていない言語に設定されている場合、英語は Deadline Cloud モニターのデフォルト言語になります。

ジョブバンドルを送信する

AWS Deadline Cloud Monitor デスクトップアプリケーションから直接ジョブバンドルを送信できます。ジョブバンドルは、Deadline Cloud にジョブを送信するために必要なファイルと情報を含むディレクトリです。サンプルジョブバンドルについては、GitHub の [deadline-cloud-samples](#) リポジトリを参照してください。

ジョブバンドルを送信するには

- Deadline Cloud Monitor デスクトップアプリケーションで、ファイル、送信ジョブバンドルを選択します。この機能は Linux Appliance または MacOS x64 ビルドでは使用できません。

Deadline Cloud でキューとフリートの詳細を表示する

Deadline Cloud モニターを使用して、ファーム内のキューとフリートの設定を表示できます。モニターを使用して、キュー内のジョブまたはフリート内のワーカーのリストを表示することもできます。

キューとフリートの詳細を表示するには、アクセスVIEWING許可が必要です。詳細が表示されない場合は、管理者に連絡して正しいアクセス許可を取得してください。

キューの詳細を表示するには

1. [Deadline Cloud モニターを開く](#)。
2. ファームのリストから、関心のあるキューを含むファームを選択します。
3. キューのリストで、キューを選択して詳細を表示します。2 つ以上のキューの設定を比較するには、複数のチェックボックスをオンにします。
4. キュー内のジョブのリストを表示するには、キューのリストから、または詳細パネルからキュー名を選択します。

モニターが既に関いている場合は、左側のナビゲーションペインのキューリストからキューを選択できます。

フリートの詳細を表示するには

1. [Deadline Cloud モニターを開く](#)。
2. ファームのリストから、関心のあるフリートを含むファームを選択します。
3. Farm リソースで、フリートを選択します。
4. フリートのリストで、フリートを選択してその詳細を表示します。2 つ以上のフリートの設定を比較するには、複数のチェックボックスをオンにします。
5. フリート内のワーカーのリストを表示するには、フリートのリストから、または詳細パネルからフリート名を選択します。

モニターが既に関いている場合は、左側のナビゲーションペインのフリートリストからフリートを選択できます。

Deadline Cloud でジョブ、ステップ、タスクを管理する

キューを選択すると、Deadline Cloud モニターのジョブモニターセクションに、そのキューのジョブ、ジョブのステップ、各ステップのタスクが表示されます。ジョブ、ステップ、またはタスクを選択すると、アクションメニューを使用してそれぞれを管理できます。

ジョブモニターを開くには、ステップに従って [キューを表示し Deadline Cloud でキューとフリートの詳細を表示する](#)、使用するジョブ、ステップ、またはタスクを選択します。

ジョブ、ステップ、タスクの場合は、以下を実行できます。

- ステータスを Requeued、Succeeded、Failed、Canceled に変更します。
- 処理された出力をジョブ、ステップ、またはタスクからダウンロードします。
- ジョブ、ステップ、またはタスクの ID をコピーします。

選択したジョブでは、次のことができます。

- ジョブをアーカイブします。
- 名前、説明、優先度、最大ワーカー数など、ジョブプロパティを変更します。
- ステップ間の依存関係を表示します。

- ジョブのパラメータを使用して追加の詳細を表示します。
- ジョブを再送信します。

詳細については、[Deadline Cloud でのジョブの詳細の表示と管理](#) を参照してください。

ステップごとに、次のことができます。

- ステップの依存関係を表示します。ステップの依存関係は、ステップを実行する前に完了する必要があります。

詳細については、「[Deadline Cloud でステップを表示する](#)」を参照してください。

タスクごとに、次のことができます。

- タスクのログを表示します。
- タスクパラメータを表示します。

詳細については、「[Deadline Cloud でタスクを表示する](#)」を参照してください。

Deadline Cloud でのジョブの詳細の表示と管理

Deadline Cloud モニターのジョブモニターページには、次の情報が表示されます。

- ジョブの進行状況の全体ビュー。
- ジョブを構成するステップとタスクのビュー。

リストからジョブを選択してジョブのステップのリストを表示し、ステップのリストからステップを選択してジョブのタスクを表示します。項目を選択したら、その項目のアクションメニューを使用して詳細を表示できます。

ジョブの詳細を表示するには

1. 「」の手順に従ってキューを表示します[Deadline Cloud でキューとフリートの詳細を表示する](#)。
2. ナビゲーションペインで、ジョブを送信したキューを選択します。
3. 次のいずれかの方法を使用してジョブを選択します。

- a. ジョブリストから、詳細を表示するジョブを選択します。
- b. 検索フィールドから、ジョブ名やジョブを作成したユーザーなど、ジョブに関連付けられたテキストを入力します。表示される結果から、表示するジョブを選択します。

ジョブの詳細には、ジョブのステップと各ステップのタスクが含まれます。アクションメニューを使用して、以下を実行できます。

- ジョブのステータスを変更します。ジョブの再キューイング、再開、停止、成功としてマーク、失敗としてマーク、またはキャンセルを行うことができます。
- 異なるプロパティまたは設定でジョブを再送信します。
- ジョブをアーカイブします。詳細については、「[ジョブをアーカイブする](#)」を参照してください。
- ジョブのプロパティを表示および変更します。
 - ジョブのステップ間の依存関係を表示できます。
 - キュー内のジョブの優先度を変更できます。優先度の高いジョブは、優先度の低いジョブの前に処理されます。ジョブの優先度は 1~100 です。2 つのジョブの優先度が同じ場合、最も古いジョブが最初にスケジュールされます。
 - ワーカーの最大数、失敗したタスクの最大数、タスクあたりの最大再試行回数を変更できます。
- ジョブの送信時に設定されたジョブのパラメータを表示します。
- ジョブの出力をダウンロードします。ジョブの出力をダウンロードすると、ジョブのステップとタスクによって生成されたすべての出力が含まれます。
- ジョブの入出力ファイルの添付ファイルを参照します。詳細については、「[Deadline Cloud でのジョブアタッチメントの表示](#)」を参照してください。

ジョブをアーカイブする

ジョブをアーカイブするには、終了状態が、`FAILED`、`SUCCEEDED`、`SUSPENDED`または `CANCELED` である必要があります。`ARCHIVED` 状態は最終です。ジョブがアーカイブされた後は、再キューに入れたい変更したりすることはできません。

ジョブのデータは、ジョブのアーカイブの影響を受けません。非アクティブタイムアウトに達するか、ジョブを含むキューが削除されると、データは削除されます。

アーカイブされたジョブで発生するその他のこと:

- アーカイブされたジョブは Deadline Cloud モニターで非表示になります。

- アーカイブされたジョブは、削除前の 120 日間、Deadline Cloud CLI の読み取り専用状態に表示されます。

ジョブをキューに入れる

ジョブを再キューに入れると、ステップ依存関係のないすべてのタスクが に切り替わりま
すREADY。依存関係を持つステップのステータスは、復元PENDING時に READYまたは に切り替わり
ます。

- すべてのジョブ、ステップ、タスクは に切り替わりますPENDING。
- ステップに依存関係がない場合は、 に切り替わりますREADY。

ジョブを再送信する

ジョブを再度実行するが、プロパティと設定が異なる場合があります。たとえば、ジョブを送信して
テストフレームのサブセットをレンダリングし、出力を確認してから、フルフレーム範囲でジョブを
再実行できます。これを行うには、ジョブを再送信します。

ジョブを再送信すると、依存関係のない新しいタスクは になりますREADY。依存関係を持つ新しい
タスクは になりますPENDING。

- すべての新しいジョブ、ステップ、タスクは になりますPENDING。
- 新しいステップに依存関係がない場合、そのステップは になりますREADY。

ジョブを再送信するときは、ジョブが最初に作成されたときに設定可能として定義されたプロパティ
のみ変更できます。たとえば、ジョブの名前が最初に送信されたときにジョブの設定可能なプロパ
ティとして定義されていない場合、再送信時に名前を編集することはできません。

Deadline Cloud でステップを表示する

AWS Deadline Cloud モニターを使用して、処理ジョブのステップを表示します。ジョブモニ
ターのステップリストには、選択したジョブを構成するステップのリストが表示されます。ステップ
を選択すると、タスクリストにステップのタスクが表示されます。

ステップを表示するには

1. ジョブのリストを表示するには、[Deadline Cloud でのジョブの詳細の表示と管理](#)「」のステップに従います。
2. [ジョブ] リストからジョブを選択します。
3. ステップリストからステップを選択します。

アクションメニューを使用して、以下を実行できます。

- ステップのステータスを変更します。再キュー、再開、一時停止、成功としてマーク、失敗としてマーク、またはステップをキャンセルできます。
- ステップの出力をダウンロードします。ステップの出力をダウンロードすると、ステップのタスクによって生成されたすべての出力が含まれます。
- ステップの入出力ファイルの添付ファイルを参照します。詳細については、「[Deadline Cloud でのジョブアタッチメントの表示](#)」を参照してください。
- ステップの依存関係を表示します。依存関係テーブルには、選択したステップを開始する前に完了する必要があるステップのリストと、このステップの完了を待っているステップのリストが表示されます。

Deadline Cloud でタスクを表示する

AWS Deadline Cloud モニターを使用して、処理ジョブのタスクを表示します。ジョブモニターのタスクリストには、ステップリストで選択したステップを構成するタスクが表示されます。

タスクを表示するには

1. ジョブのリストを表示するには、[Deadline Cloud でのジョブの詳細の表示と管理](#)「」のステップに従います。
2. [ジョブ] リストからジョブを選択します。
3. ステップリストからステップを選択します。
4. タスクリストからタスクを選択します。

アクションメニューを使用して、以下を実行できます。

- タスクのステータスを変更します。タスクを再キューに入れる、一時停止する、成功としてマークする、失敗としてマークする、またはキャンセルすることができます。

- タスク実行を表示して、タスクの試行履歴を表示します。
- セッションログとワーカーログを表示します。詳細については、「[Deadline Cloud でセッションログとワーカーログを表示する](#)」を参照してください。
- タスクを処理したワーカーのワーカーダッシュボードを表示します。詳細については、「[ワーカーダッシュボードでワーカーの詳細を表示する](#)」を参照してください。
- タスクの作成時に設定されたパラメータを表示します。
- タスクの出力をダウンロードします。タスクの出力をダウンロードすると、選択したタスクによって生成された出力のみが含まれます。
- タスクの入出力ファイルの添付ファイルを参照します。詳細については、「[Deadline Cloud でのジョブアタッチメントの表示](#)」を参照してください。

Deadline Cloud でセッションログとワーカーログを表示する

ログは、タスクのステータスと処理に関する詳細情報を提供します。AWS Deadline Cloud モニターには、次の 2 種類のログが表示されます。

- セッションログには、以下を含むアクションのタイムラインが詳述されています。
 - アタッチメントの同期やソフトウェア環境のロードなどのセットアップアクション
 - タスクまたは一連のタスクの実行
 - ワーカーの環境をシャットダウンするなどの終了アクション

セッションには少なくとも 1 つのタスクの処理が含まれ、複数のタスクを含めることができます。セッションログには、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスタイプ、vCPU、メモリに関する情報も表示されます。セッションログには、セッションで使用されるワーカーのログへのリンクも含まれています。

- ワーカーログは、ワーカーがライフサイクル中に処理するアクションのタイムラインの詳細を提供します。ワーカーログには、複数のセッションに関する情報を含めることができます。

セッションログとワーカーログをダウンロードして、オフラインで調べることができます。

セッションログを表示するには

1. ジョブのリスト [Deadline Cloud でのジョブの詳細の表示と管理](#) を表示するには、「」のステップに従います。
2. [ジョブ] リストからジョブを選択します。

3. ステップリストからステップを選択します。
4. タスクリストからタスクを選択します。
5. アクションメニューから、ログの表示を選択します。

タイムラインセクションには、タスクのアクションの概要が表示されます。セッションで実行されているタスクをさらに表示し、セッションのシャットダウンアクションを確認するには、すべてのタスクのログを表示するを選択します。

タスクからワーカーログを表示するには

1. ジョブのリスト [Deadline Cloud でのジョブの詳細の表示と管理](#) を表示するには、「」のステップに従います。
2. [ジョブ] リストからジョブを選択します。
3. ステップリストからステップを選択します。
4. タスクリストからタスクを選択します。
5. アクションメニューから、ログの表示を選択します。
6. セッション情報を選択します。
7. ワーカーログの表示 を選択します。

フリートの詳細からワーカーログを表示するには

1. フリート [Deadline Cloud でキューとフリートの詳細を表示する](#) を表示するには、「」のステップに従います。
2. ワーカーリストからワーカー ID を選択します。
3. アクションメニューから、ワーカーログの表示を選択します。

ワーカーダッシュボードでワーカーの詳細を表示する

ワーカーダッシュボードには、タスクを処理するワーカーの詳細が表示されます。以下を確認できます。

- ワーカーのインスタンスタイプなどのメタデータ。
- ワーカーが実行したセッションアクション。

- CPU、メモリ、ディスク使用量などのワーカーのパフォーマンス。ダッシュボードには、GPU アクセラレーションインスタンスの GPU 使用率も表示されます。
- CPU、メモリ、ディスクの経時的な使用状況のグラフ。このグラフには、GPU アクセラレーションインスタンスの GPU 使用率も含まれています。
- 時間の経過に伴うディスク速度のグラフ。
- タスクのワーカーログ。

タスクからワーカーダッシュボードを表示するには

1. ジョブのリスト [Deadline Cloud でのジョブの詳細の表示と管理](#) を表示するには、「」のステップに従います。
2. [ジョブ] リストからジョブを選択します。
3. ステップリストからステップを選択します。
4. タスクリストからタスクを選択します。
5. タスクテーブルのアクションメニューから、ワーカーダッシュボードの表示を選択します。

フリートの詳細からワーカーダッシュボードを表示するには

1. フリート [Deadline Cloud でキューとフリートの詳細を表示する](#) を表示するには、「」のステップに従います。
2. ワーカーリストからワーカーを選択します。
3. アクションメニューから、ワーカーダッシュボードの表示を選択します。

ユースケース

プロビジョニング不足のインスタンスの検出

レンダリングに予想以上に時間がかかる場合、ワーカーダッシュボードは、インスタンスがワークロードに適したサイズであるかどうかを判断するのに役立ちます。多くのレンダーラーでは 100% の vCPU 使用率が正常ですが、最大容量に近いメモリ使用率が一貫して高く、ディスク容量使用率が高くなると、インスタンスのプロビジョニングが不足している可能性があります。GPU アクセラレーションワークロードの場合、GPU 使用率が一貫して高い場合も、追加の GPU 容量が必要であることを示している可能性があります。このような場合、フリートのインスタンス設定をアップグレードすると、レンダリングエラーが減少し、レンダリング時間が大幅に短縮されます。ただし、最適なバランスを見つけるには、アップグレード後もワーカーのパフォーマンスをモニタリングし続けること

が重要です。アップグレードが過度に積極的に行われると、オーバープロビジョニングによって不要なコストが発生する可能性があります。

過剰にプロビジョニングされたインスタンスの検出

タスクが正常に完了しても、コストを最適化する機会があるかもしれません。ワーカーダッシュボードでは、ワークロードが必要とするよりも多くのコンピューティング能力に料金を支払っているかどうかを確認できます。ワーカーの平均 vCPU 使用率が低く、メモリ使用率が最小限で、未使用のディスク容量が過剰である場合は、フリートのインスタンス設定をダウンサイズできます。

失敗したタスクのトラブルシューティング

失敗したタスクを調査する場合、ワーカーダッシュボードは貴重な診断ツールとして機能します。ピーク時のメモリ使用量とディスク容量使用率に特に注意してください。これらのメトリクスが 100% に近づいた場合、または 100% に達した場合は、タスクの失敗の根本原因である可能性があります。このようなリソースの枯渇は、現在のインスタンスにワークロードを効果的に処理する容量がないことを示します。このような場合、メモリまたはディスク容量を増やしたインスタンスをプロビジョニングすると、タスクが正常に完了するのに役立ちます。

最適なインスタンス使用率

vCPU 使用率

ターゲット範囲: 70~90%

- 70% 未満: コンピューティングリソースの使用率が低い可能性があります。つまり、ワークロードのニーズよりも多くの CPU に支払うこととなります。
- 70~90%: ボトルネックに遭遇することなくリソースを効率的に使用する最適な範囲
- 一貫して 100% の場合: レンダリングが遅くなる可能性のある CPU ボトルネックを示している可能性があります

一部のレンダリングタスクは、他のタスクよりも CPU 負荷が高く、vCPU 使用率が 100% でも問題にならない場合があることに注意してください。リアルタイムビジュアライゼーションタスクは、より一貫した CPU 使用率を示す可能性があります。計算要件が変化するタスクのパターンは異なる場合があります。

メモリ使用率

ターゲット範囲: 70~85%

- 50% 未満: ワークロードに対してサイズが大きすぎる可能性のあるインスタンス
- 70~85%: スパイクに十分なヘッドルームを備えた最適な使用率
- 90% を超える: パフォーマンスの低下またはout-of-memoryエラーのリスク

メモリ要件は、シーンの複雑さ、テクスチャ解像度、シミュレーションデータによって大きく異なる場合があります。時間の経過に伴うメモリの傾向をモニタリングすることは、ワークロードがメモリ要件で増加しているかどうかを特定する上で重要です。

ディスクスペース使用率

ターゲット範囲: 60~80%

- 40% 未満: おそらく過剰にプロビジョニングされたストレージ
- 60 ~ 85%: 一時ファイルとキャッシュ用のスペースがあり、使用率が良い
- 85% 以上: 大規模なレンダリング中にスペースが不足するリスク

ディスク I/O パフォーマンスは、特にレンダリング中に大きなテクスチャまたはキャッシュファイルを読み書きするワークロードの場合、容量と同じくらい重要になる可能性があることに注意してください。

Deadline Cloud で完成した出力をダウンロードする

ジョブが完了したら、Deadline Cloud AWS モニターを使用して結果をワークステーションにダウンロードできます。出力ファイルは、ジョブの作成時に指定した名前と場所とともに保存されます。

出力ファイルは無期限に保存されます。ストレージコストを削減するには、キューの Amazon S3 バケットの S3 ライフサイクル設定を作成することを検討してください。Amazon S3 詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[ストレージライフサイクルの管理](#)」を参照してください。

ジョブ、ステップ、またはタスクの完成した出力をダウンロードするには

1. ジョブのリストを表示するには、[Deadline Cloud でのジョブの詳細の表示と管理](#)「」のステップに従います。
2. 出力をダウンロードするジョブ、ステップ、またはタスクを選択します。
 - ジョブを選択した場合は、そのジョブのすべてのステップで、すべてのタスクのすべての出力をダウンロードできます。

- ステップを選択すると、そのステップのすべてのタスクのすべての出力をダウンロードできます。
 - タスクを選択すると、その個々のタスクの出力をダウンロードできます。
3. Actions メニューから、Download output を選択します。
 4. 出力は、ジョブの送信時に設定された場所にダウンロードされます。

Note

メニューを使用した出力のダウンロードは、現在 Windows および macOS でのみサポートされています。Linux、Mac、出力メニュー項目のダウンロードを選択すると、レンダリングされた出力のダウンロードに使用できる AWS CLI コマンドがウィンドウに表示されます。

すべての出力ではなく個々のファイルを参照して選択的にダウンロードするには、「」を参照してください [Deadline Cloud でのジョブアタッチメントの表示](#)。

Deadline Cloud でのジョブアタッチメントの表示

Deadline Cloud モニターのジョブアタッチメントブラウザを使用して、ジョブの入出力アタッチメントのファイル構造を確認します。ファイルの選択的なダウンロード、フォルダ階層のナビゲート、入力または出力によるフィルタリングを行うことができます。サポートされているファイルタイプでは、ダウンロードせずにコンテンツをインラインでプレビューすることもできます。

添付ファイルブラウザには、入力ファイル (ジョブで送信) と出力ファイル (処理中にワーカーが生成) の両方が表示されます。ビューをフィルタリングして、入力のみ、出力のみ、またはすべてのファイルをまとめて表示できます。

アタッチメントブラウザを開く

ジョブ、ステップ、またはタスクから添付ファイルブラウザを開くことができます。表示されるファイルの範囲は、ブラウザを開くレベルによって異なります。

ジョブの添付ファイルを参照するには

1. ジョブのリストを表示するには、[Deadline Cloud でのジョブの詳細の表示と管理](#)「」のステップに従います。
2. 添付ファイルを参照するジョブを選択します。

3. アクションメニューから、添付ファイルを参照を選択します。

特定のステップまたはタスクの添付ファイルを表示するには、ステップまたはタスクを選択し、アクションメニューから添付ファイルを参照を選択します。ステップまたはタスクレベルで参照すると、ブラウザにはステップまたはタスクが生成した出力ファイルのみが表示されます。

ジョブ、ステップ、またはタスクのコンテキストメニューを開き、添付ファイルの参照を選択することもできます。

ファイルのナビゲーションとフィルタリング

添付ファイルブラウザには、元のファイルパスで整理された展開可能なツリー構造にファイルが表示されます。次のコントロールを使用して移動します。

- セグメント制御 – すべて、出力、または入力ファイルの表示を切り替えます。
- テキストフィルター – ファイルリストを名前でフィルタリングして、特定のファイルを見つけます。
- フォルダの展開 – フォルダを選択して、その内容を展開または折りたたみます。

ファイルのダウンロード

添付ファイルブラウザから個々のファイルまたはファイルの選択をダウンロードできます。

選択したファイルをダウンロードするには

1. 添付ファイルブラウザで、ダウンロードするファイルまたはフォルダの横にあるチェックボックスをオンにします。フォルダを選択すると、そのフォルダ内のすべてのファイルが選択されます。
2. [ダウンロード] を選択します。
3. ダウンロード方法を選択します。
 - AWS Command Line Interface (AWS CLI) – ストレージプロファイルパスにファイルをダウンロードし、元のディレクトリ構造を維持します。この方法は、大きなファイルや、元の場所にファイルが必要な場合に使用します。
 - ブラウザのダウンロード – 選択したファイルを ZIP アーカイブとしてブラウザに直接ダウンロードします。少数のファイルにすばやくアクセスするには、この方法を使用します。合計サイズが 500 MB を超えると警告が表示されます。

Note

Deadline Cloud Monitor デスクトップアプリケーション (Windows、macOS、および Linux) では、AWS CLI ダウンロードはコマンドをコピーすることなく直接実行されます。デスクトップアプリケーションは、元のストレージプロファイルパスへのファイルのダウンロードもサポートしています。

ファイルのプレビュー

サポートされているファイルタイプでは、添付ファイルブラウザーはファイルをダウンロードせずにインラインプレビューを表示できます。

ファイルをプレビューするには

1. 添付ファイルブラウザーで、テーブル内のファイル名を選択します。
2. 側面で開く詳細パネルを表示します。パネルには、ファイルサイズ、パス、変更日などのメタデータが表示されます。サポートされているファイルタイプでは、インラインプレビューが自動的にロードされます。
3. (オプション) 全画面プレビューを開くには、ファイル行にカーソルを合わせたときに表示される展開アイコンを選択します。

次のファイルタイプはインラインプレビューをサポートしています。

- イメージ: PNG、JPG、JPEG、GIF、BMP、WebP、SVG、ICO
- テキスト:
TXT、LOG、JSON、XML、YAML、YML、CSV、MD、INI、CFG、CONF、SH、BAT、PY、JS、TS、T
- 動画: MP4、WebM、OGG
- オーディオ: MP3、WAV、OGG、FLAC、AAC、WebM

Deadline Cloud Monitor デスクトップのデプロイとワークフローを自動化する

AWS Deadline Cloud モニターデスクトップアプリケーションには、管理者がユーザーのプロファイルを設定アップするために使用できるコマンドラインインターフェイス (CLI) と、アーティストや

デベロッパーがモニターをワークステーションの自動ワークフローに統合するために使用できるコマンドラインインターフェイス (CLI) が含まれています。

Deadline Cloud Monitor 実行可能ファイルの検索

CLI コマンドを使用するには、ターミナルから Deadline Cloud Monitor 実行可能ファイルを実行します。デフォルトのインストール場所は、オペレーティングシステムとインストール方法によって異なります。

Windows

```
%LOCALAPPDATA%\DeadlineCloudMonitor\DeadlineCloudMonitor.exe
```

macOS

```
/Applications/DeadlineCloudMonitor.app/Contents/MacOS/DeadlineCloudMonitor
```

Linux (deb または RPM パッケージ)

```
/usr/bin/deadline-cloud-monitor
```

Linux (Applmage)

Applmage ファイルは、ダウンロードした場所から直接実行します。

次の例では、をオペレーティングシステムの実行可能ファイルへのフルパスDeadlineCloudMonitorに置き換えます。

ユーザーアクセスを合理化するためのプロファイルの設定

管理者は create-profile コマンドを使用して、ユーザーの Deadline Cloud Monitor プロファイルを作成します。このコマンドは、ユーザーが追加の設定やプロファイルを選択せずにモニターを開いてログインし、作業を開始できるようにプロファイルを設定します。

create-profile コマンドは、次のフラグを受け入れます。

- `--enable-auto-login` – アプリケーションの起動時に、最後に使用したプロファイルで自動的にログインするようにモニタを設定します。

- `--set-as-deadline-default` – Deadline Cloud 送信者、Deadline CLI、Deadline Cloud GUI アプリケーションなど、Deadline Cloud ツールのデフォルトとしてプロファイルを設定します。このフラグは () には AWS Command Line Interface 影響しませんAWS CLI。

両方のフラグを有効にすると、ユーザーはモニターを開き、他の設定やプロファイルの選択を必要とせずに自動的にログインします。

プロファイルを作成するには

次のコマンドを実行し、プレースホルダー値をモニターの詳細に置き換えます。

```
DeadlineCloudMonitor create-profile \  
  --profile profile-name \  
  --monitor-id monitor-id \  
  --monitor-url https://monitorName.region.deadlinecloud.amazonaws.com \  
  --enable-auto-login \  
  --set-as-deadline-default
```

コマンドはプロファイルを作成し、ユーザーのワークステーションの Deadline Cloud 設定ファイルに設定を書き込みます。モニター URL は の形式である必要があります `https://monitorName.region.deadlinecloud.amazonaws.com`。

Note

`create-profile` コマンドは、プロファイルの作成後に終了します。新しいプロファイルでモニターを開くには、`login` コマンドを実行するか、Deadline Cloud Monitor デスクトップアプリケーションを開きます。

Deadline Cloud モニターをワークフローに統合する

`login`、`logout`、および `handle-url` コマンドを使用して、Deadline Cloud モニターをワークステーションのスクリプトと自動ワークフローに統合します。

ログインとログアウト

`login` および `logout` コマンドを使用して、ワークフローの一部として認証を制御します。たとえば、ジョブを送信するスクリプトは、`login` コマンドを使用して、送信を開始する前にユーザーが認証されるようにできます。

login コマンドを使用すると、モニタは指定されたプロファイルに直接開き、プロファイル選択画面はスキップされます。認証が完了すると、モニターはシステムトレイに最小化され、ワークフローを続行できます。モニタが指定されたプロファイルに対して既に実行されている場合、既存のウィンドウは新しいインスタンスを起動する代わりにフォアグラウンドに表示されます。

プロファイルにログインするには

`profile-name` を Deadline Cloud モニタープロファイルの名前に置き換えて、次のコマンドを実行します。

```
DeadlineCloudMonitor login --profile profile-name
```

プロファイルからログアウトするには

次のコマンドを実行して、プロファイルの認証情報をクリアし、そのプロファイルの実行中のモニターインスタンスが終了するようにシグナルを送信します。

```
DeadlineCloudMonitor logout --profile profile-name
```

モニターを特定のページに開く

handle-url コマンドを使用して、Deadline Cloud モニターを特定のページに開きます。このコマンドは、スクリプトがジョブの作成などのアクションを実行し、自動的にモニターを開いて結果を表示する場合に便利です。たとえば、スクリプトがジョブを送信した後、スクリプトは handle-url を呼び出してモニターをジョブの詳細ページに直接開くことができます。

deadline-cloud-monitor:// URL を企業ウェブサイト、Wiki、またはタスクトラッカーのリンクとして使用して、ユーザーがモニターを特定のページに直接開くようにすることもできます。

URL は、launch コマンドで deadline-cloud-monitor:// プロトコルスキームを使用します。URL には、プロファイル名と開くモニターページの URL が含まれます。

特定のページにモニターを開くには

monitor*monitor-page-url* を URL エンコードされたモニターページの URL に置き換え、*profile-name* をプロファイル名に置き換えて、次のコマンドを実行します。

```
DeadlineCloudMonitor handle-url --url "deadline-cloud-monitor://launch?url=monitor-  
page-url&profile=profile-name"
```

Deadline クラウドファーム

Deadline Cloud ファームを使用すると、ユーザーとプロジェクトリソースを管理できます。ファームは、プロジェクトリソースが配置されているです。ファームはキューとフリートで構成されます。キューは、送信されたジョブが配置され、レンダリングがスケジュールされる場所です。フリートは、タスクを実行してジョブを完了するワーカーノードのグループです。ファームを作成したら、プロジェクトのニーズに合わせてキューとフリートを作成できます。

ファームを作成する

1. [Deadline Cloud コンソール](#)から、ダッシュボードに移動を選択します。
2. Deadline Cloud ダッシュボードの Farms セクションで、Actions → Create farm を選択します。
 - または、左側のパネルでファームやその他のリソースを選択し、ファームの作成を選択します。
3. ファームの名前を追加します。
4. 説明 にファームの説明を入力します。明確な説明は、ファームの目的をすばやく特定するのに役立ちます。
5. (オプション) デフォルトでは、データはセキュリティのために が AWS 所有および管理するキーで暗号化されます。暗号化設定をカスタマイズ (詳細) を選択して、既存のキーを使用するか、管理する新しいキーを作成できます。

チェックボックスを使用して暗号化設定をカスタマイズする場合は、AWS KMS ARN を入力するか、新しい KMS キーの作成 AWS KMS を選択して新しい を作成します。

6. (オプション) Cost scale factor に値を入力して、Usage Explorer と予算マネージャーでのコストの表示方法を調整します。1 未満の値は割引を表し、1 より大きい値はプレミアムを表し、1 (デフォルト) はコストを変更しません。詳細については、「[コストスケール係数](#)」を参照してください。
7. (オプション) 新しいタグを追加を選択して、ファームに 1 つ以上のタグを追加します。
8. Create farm を選択します。作成後、ファームが表示されます。

Deadline クラウドキュー

キューは、ジョブを管理および処理するファームリソースです。

キューを使用するには、モニターとファームが既にセットアップされている必要があります。

トピック

- [キューを作成する](#)
- [キュー環境を作成する](#)
- [キューとフリートに関連付ける](#)

キューを作成する

1. [Deadline Cloud コンソール](#)ダッシュボードから、キューを作成するファームを選択します。
 - または、左側のパネルで Farms およびその他のリソースを選択し、キューを作成するファームを選択します。
2. 「キュー」タブで「キューの作成」を選択します。
3. キューの名前を入力します。
4. 説明 にキューの説明を入力します。説明は、キューの目的を特定するのに役立ちます。
5. ジョブアタッチメントでは、新しい Amazon S3 バケットを作成するか、既存の Amazon S3 バケットを選択できます。
 - a. 新しい Amazon S3 バケットを作成するには
 - i. 新しいジョブバケットの作成を選択します。
 - ii. バケットの名前を入力します。バケット に名前を付けることをお勧めします `deadlinecloud-job-attachments-[MONITORNAME]`。
 - iii. ルートプレフィックスを入力して、キューのルートの場所を定義または変更します。
 - b. 既存の Amazon S3 バケットを選択するには
 - i. 既存の S3 バケットを選択 > S3 を参照を選択します。
 - ii. 使用可能なバケットのリストからキューの S3 バケットを選択します。
6. (オプション) キューをカスタマーマネージドフリートに関連付けるには、カスタマーマネージドフリートとの関連付けを有効にするを選択します。

7. カスタマーマネージドフリートとの関連付けを有効にする場合は、次の手順を実行する必要があります。

⚠ Important

run-as 機能用にユーザーとグループを指定することを強くお勧めします。そうしないと、ジョブはワーカーのエージェントができることをすべて実行できるため、ファームのセキュリティ体制が低下します。潜在的なセキュリティリスクの詳細については、[「ユーザーおよびグループとしてジョブを実行する」](#)を参照してください。

- a. ユーザーとして実行の場合:

キューのジョブの認証情報を指定するには、キュー設定ユーザーを選択します。

または、独自の認証情報の設定をオプトアウトし、ワーカーエージェントユーザーとしてジョブを実行するには、ワーカーエージェントユーザーを選択します。

- b. (オプション) ユーザー認証情報として実行 で、ユーザー名とグループ名を入力して、キューのジョブの認証情報を指定します。

Windows フリートを使用している場合は、ユーザーとして実行のパスワードを含むシークレットを作成 AWS Secrets Manager する必要があります。パスワードを持つ既存のシークレットがない場合は、シークレットの作成を選択して Secrets Manager コンソールを開き、シークレットを作成します。詳細については、「Deadline Cloud Developer Guide」の[「Manage access to Windows job user secrets」](#)を参照してください。

8. 予算を必須にすることは、キューのコストを管理するのに役立ちます。予算を必要としないか、予算が必要かを選択します。
9. キューには、ユーザーに代わって Amazon S3 にアクセスするためのアクセス許可が必要です。新しいサービスロールを作成することも、既存のサービスロールを使用することもできます。既存のサービスロールがない場合は、新しいサービスロールを作成して使用します。
 - a. 既存のサービスロールを使用するには、サービスロールの選択を選択し、ドロップダウンからロールを選択します。
 - b. 新しいサービスロールを作成するには、新しいサービスロールを作成して使用し、ロール名と説明を入力します。
10. (オプション) キュー環境の環境変数を追加するには、新しい環境変数を追加を選択し、追加する各変数の名前と値を入力します。

11. (オプション) キューに 1 つ以上のタグを追加するには、新しいタグの追加を選択します。
12. デフォルトのcondaキュー環境を作成するには、チェックボックスをオンにしたままにします。キュー環境の詳細については、「[キュー環境の作成](#)」を参照してください。カスターマネージドフリートのキューを作成する場合は、チェックボックスをオフにします。
13. [キューの作成] を選択します。

キュー環境を作成する

キュー環境は、フリートワーカーを設定する一連の環境変数とコマンドです。キュー環境を使用して、ソフトウェアアプリケーション、環境変数、その他のリソースをキュー内のジョブに提供できます。

キューを作成するときは、デフォルトのcondaキュー環境を作成するオプションがあります。この環境では、サービスマネージドフリートがパートナー DCC アプリケーションとレンダラーのパッケージにアクセスできます。デフォルトの環境 詳細については、「」を参照してください [デフォルトのcondaキュー環境](#)。

キュー環境を追加するには、コンソールを使用するか、json または YAML テンプレートを直接編集します。この手順では、コンソールを使用して環境を作成する方法について説明します。

1. キューにキュー環境を追加するには、キューに移動し、キュー環境タブを選択します。
2. アクションを選択し、フォームを使用して新しい を作成します。
3. キュー環境の名前と説明を入力します。
4. 新しい環境変数を追加を選択し、追加する各変数の名前と値を入力します。
5. (オプション) キュー環境の優先度を入力します。優先度は、このキュー環境がワーカーで実行される順序を示します。優先度の高いキュー環境が最初に実行されます。
6. キュー環境の作成 を選択します。

デフォルトのcondaキュー環境

サービスマネージドフリートに関連付けられたキューを作成する場合、がジョブの仮想環境にパッケージをダウンロードしてインストール [conda](#) するデフォルトのキュー環境を追加するオプションがあります。

Deadline Cloud [コンソール](#) でデフォルトのキュー環境を追加すると、環境が自動的に作成されます。AWS CLI や を使用して別の方法でキューを追加する場合は CloudFormation、キュー環境を

自分で作成する必要があります。環境に正しいコンテンツがあることを確認するには、GitHub のキュー環境テンプレート YAML ファイルを参照してください。デフォルトのキュー環境の内容については、GitHub の [デフォルトのキュー環境 YAML ファイル](#) を参照してください。

GitHub には、独自のニーズの出発点として使用できる他の [キュー環境テンプレート](#) があります。

Conda はチャンネルからのパッケージを提供します。チャンネルは、パッケージが保存される場所です。Deadline Cloud は、パートナー DCC アプリケーションとレンダラーをサポートする conda パッケージ `deadline-cloud` をホストするチャンネルを提供します。以下の各タブを選択すると、Linux または で使用できるパッケージが表示されます Windows。

Linux

- Autodesk Arnold for Cinema 4D
 - `cinema4d-c4dtoa=2025`
- Autodesk Arnold for Maya
 - `maya-mtoa=2024.5.3`
 - `maya-mtoa=2025.5.4`
 - `maya-mtoa=2026.5.5`
- Autodesk Maya
 - `maya=2024`
 - `maya=2025`
 - `maya=2026`
 - `maya-openjd`
- Autodesk VRED
 - `vredcore=2025`
 - `vredcore=2026`
- ブレンダー
 - `blender=3.6`
 - `blender=4.2`
 - `blender=4.5`
 - `blender=5.0`
 - `blender=5.1`
 - `blender-openjd`

- Chaos V-Ray for Maya
 - `maya-vray=2025.7`
 - `maya-vray=2026.7`
- Foundry Nuke
 - `nuke=15`
 - `nuke=16`
 - `nuke=17`
 - `nuke-openjd`
- Maxon シネマ 4D
 - `cinema4d=2025`
 - `cinema4d=2026`
 - `cinema4d-openjd`
- Maxon Redshift for Maya
 - `maya-redshift=2025.4`
 - `maya-redshift=2026.2`
- SideFX Houdini
 - `houdini=19.5`
 - `houdini=20.0`
 - `houdini=20.5`
 - `houdini=21.0`
 - `houdini-openjd`

Windows

- Adobe After Effects
 - `aftereffects=24.6`
 - `aftereffects=25.1`
 - `aftereffects=25.2`
 - `aftereffects=25.6`
 - `aftereffects=26.0`

- Autodesk Arnold for Cinema 4D

- cinema4d-c4dtoa=2025
- cinema4d-c4dtoa=2026
- Maxon シネマ 4D
 - cinema4d=2024
 - cinema4d=2025
 - cinema4d=2026
 - cinema4d-openjd
- Unreal Engine
 - unrealengine=5.4
 - unrealengine=5.5
 - unrealengine=5.6
 - unrealengine=5.7
 - unrealengine-openjd

Note

Cinema 4D の場合、Linuxconda パッケージは素材 3D マテリアルをサポートしていません。このマテリアルを持つジョブは、次のいずれかのエラーで失敗します。

```
Commandline: ./modules/io_substance/source/substance_framework/src/details/detailsengine.cpp:794: SubstanceAir::Details::Engine::Context::Context(SubstanceAir::Details::Engine&, SubstanceAir::RenderCallbacks*): Assertion `res==0' failed.
```

```
/home/job-user/.conda/envs/<hash>/Lib/deadline/cinema4d_adaptor/Cinema4DAdaptor/adaptor.sh: line 44: 10832 Segmentation fault (core dumped) $C4DEXE ${ARGS[*]}
```

Windows 代わりに、素材マテリアルを含むジョブを に送信することをお勧めします。の Cinema 4D 2025.3.3 ではLinux、グローバル化されたアセットパスがセグメンテーションの障害を引き起こす可能性があります。したがって、Linuxconda パッケージには、代わりに Redshift 2025.6.0 を含む Cinema 4D 2025.3.1 が含まれています。Cinema 4D 2025.3.3 の機能またはバグ修正が必要な場合は、Cinema 4D 2026 にアップグレードするか、Windows 代わりにそれらのジョブを に送信するという 2 つのオプションをお勧めします。

Cinema 4D OpenJD では、タイムアウトの問題を防ぐために、デフォルトの 2 日間のタイムアウトを使用する代わりに、タスク実行タイムアウトを予想レンダリング時間の 2 倍に設定することをお勧めします。

デフォルトconda環境のキューにジョブを送信すると、環境はジョブに 2 つのパラメータを追加します。これらのパラメータは、タスクが処理される前にジョブの環境を設定するために使用するcondaパッケージとチャンネルを指定します。パラメータは次のとおりです。

- CondaPackages – blender=3.6や など、[パッケージ一致仕様](#)のスペース区切りリストnumpy>1.22。仮想環境の作成をスキップするには、デフォルトは空です。
- CondaChannels – deadline-cloud、conda-forgeなどの[condaチャンネル](#)のスペース区切りリストs3://*amzn-s3-demo-bucket*/conda/channel。デフォルトは `s3://amzn-s3-demo-bucket/conda/channel` です。これは deadline-cloud、パートナー DCC アプリケーションとレンダラーを提供するサービスマネージドフリートで使用できるチャンネルです。

統合された送信者を使用して DCC から Deadline Cloud にジョブを送信すると、送信者は DCC アプリケーションと送信者に基づいて CondaPackagesパラメータの値を入力します。たとえば、Blender を使用している場合、CondaPackageパラメータは `blender=3.6.* blender-openjd=0.4.*` に設定されます。

上記の表に記載されているバージョンにのみ送信をピン留めすることをお勧めします。例: blender=3.6。パッチリリースは使用可能なパッケージに影響するため、major.minor バージョンへのピン留めをお勧めします。たとえば、3.6.17 Blender をリリースすると、3.6.16 Blender は配布されなくなります。blender=3.6.16 にピン留めされた送信は失敗します。blender=3.6 にピン留めすると、最新の分散パッチバージョンを取得でき、ジョブは影響を受けません。デフォルトでは、DCC 送信者は、アンサンブル = 3.6 などのパッチ番号を除き、上記の表にリストされている現在のバージョンに固定されます。

キューとフリートを関連付ける

ジョブを処理するには、キューをフリートに関連付ける必要があります。1 つのフリートを複数のキューに、1 つのキューを複数のフリートに関連付けることができます。フリートを複数のキューに関連付けると、それらの間でワーカーが均等に分割されます。同様に、キューを複数のフリートに関連付けると、それらのフリート間でジョブが均等に分散されます。

Note

待機と保存を使用するには、待機と保存のインスタンスタイプを使用するフリートにのみキューを関連付けることをお勧めします。キューを複数のフリートに関連付け、それらのフリートのいずれかがスポットインスタンスタイプまたはオンデマンドインスタンスタイプを使用する場合、フリートは待機してインスタンスを保存しながらジョブを処理しない可能性があります。

既存のキューを既存のフリートに関連付けるには、次の手順を実行します。

1. Deadline Cloud ファームから、フリートに関連付けるキューを選択します。キューが表示されます。
2. キューに関連付けるフリートを選択するには、フリートの関連付けを選択します。
3. フリートの選択ドロップダウンを選択します。使用可能なフリートのリストが表示されます。
4. 使用可能なフリートのリストから、キューに関連付けるフリートの横にあるチェックボックスをオンにします。
5. 関連付ける を選択してください。これで、フリートの関連付けステータスはアクティブになります。

キューフリートの関連付けを停止する

キューフリートの関連付けを停止するには、次の手順を実行します。

1. キューから、関連付けられたフリートタブを選択します。
2. キューとの関連付けを停止するフリートのチェックボックスをオンにします。
3. Actions ドロップダウンから、Eventual stop または Immediate stop を選択します。

関連付けが停止する前にジョブの処理を完了するには、Eventual stop を選択します。ジョブの処理をすぐに停止するには、即時停止を選択します。

4. 確認ウィンドウで、**confirm** 「」と入力し、「停止」を選択します。
5. (オプション) キューからフリートの関連付けを解除するには、次の手順を実行します。
 - a. 関連付けステータスが Stopped に変わるまで待ちます。
 - b. 関連付けが停止した後、まだの場合は、フリートのチェックボックスをオンにします。
 - c. Actions ドロップダウンから、フリートの関連付けを解除を選択します。

- d. 確認ウィンドウで、関連付け解除を選択します。

キューフリートの関連付けを再アクティブ化する

キューフリートの関連付けを再アクティブ化するには、次の手順を実行します。

1. キューから、関連付けられたフリートタブを選択します。
2. キューフリートの関連付けを再アクティブ化するフリートのチェックボックスをオンにします。
3. Actions ドロップダウンから、Start を選択します。関連付けステータスがアクティブに変わります。

Deadline クラウドフリート

このセクションでは、Deadline Cloud のサービスマネージドフリートとカスターマネージドフリート (CMF) を管理する方法について説明します。

2 種類の Deadline Cloud フリートを設定できます。

- サービスマネージドフリートは、Deadline Cloud によってデフォルト設定が提供されているワーカーのフリートです。これらのデフォルト設定は、効率的で費用対効果が高いように設計されています。
- カスターマネージドフリート (CMFs) を使用すると、処理パイプラインを完全に制御できます。CMF は、AWS インフラストラクチャ内、オンプレミス、または同じ場所にあるデータセンター内に配置できます。CMFs には、フリート内のワーカーのプロビジョニング、オペレーション、管理、廃止が含まれます。

フリートを複数のキューに関連付けると、それらのキュー間でワーカーが均等に分割されます。

トピック

- [サービスマネージドフリート](#)
- [カスターマネージドフリート](#)
- [自動スケーリング設定](#)

サービスマネージドフリート

サービスマネージドフリート (SMF) は、Deadline Cloud によってデフォルト設定が提供されているワーカーのフリートです。これらのデフォルト設定は、効率的で費用対効果が高いように設計されています。

一部のデフォルト設定は、ワーカーとタスクが実行できる時間を制限します。ワーカーは 7 日間のみ実行でき、タスクは 5 日間のみ実行できます。制限に達すると、タスクまたはワーカーは停止します。この場合、ワーカーまたはタスクが実行されていた作業が失われる可能性があります。これを回避するには、ワーカーとタスクをモニタリングして、最大期間の制限を超えないようにします。ワーカーのモニタリングの詳細については、「」を参照してください [Deadline Cloud モニターの使用](#)。

サービスマネージドフリートを作成する

サービスマネージドフリートには、スポット、オンデマンド、wait-and-saveの3種類のインスタンスオプションを選択できます。スポットインスタンスは予約されていない容量であり、割引価格で使用できますが、オンデマンドリクエストによって中断される可能性があります。オンデマンドインスタンスの料金は2番目で、長期的なコミットメントはなく、中断されません。Wait-and-saveは、ジョブのスケジュールを遅らせてコストを削減し、オンデマンドリクエストやスポットリクエストによって中断できます。

1. [Deadline Cloud コンソール](#)から、フリートを作成するファームに移動します。
2. フリートタブを選択し、フリートの作成を選択します。
3. フリートの名前を入力します。
4. (オプション) [説明] を入力します。明確な説明は、フリートの目的をすばやく特定するのに役立ちます。
5. サーマネージドフリートタイプを選択します。
6. フリートのスポット、オンデマンド、またはインスタンスマーケットの待機と保存オプションを選択します。デフォルトでは、フリートはスポットオプションを使用します。
7. フリートのサービスアクセスの場合は、既存のロールを選択するか、新しいロールを作成します。サービスロールは、フリート内のインスタンスに認証情報を提供し、ジョブを処理するアクセス許可と、ログ情報を読み取れるようにモニター内のユーザーに付与します。
8. [次へ] を選択します。
9. CPU 専用インスタンスまたは GPU アクセラレーションインスタンスから選択します。GPU アクセラレーションインスタンスはジョブをより迅速に処理できる場合がありますが、コストがかかる場合があります。
10. ワーカーのオペレーティングシステムを選択します。デフォルトのままLinuxにするか、 を選択できますWindows。
11. (オプション) GPU アクセラレーションインスタンスを選択した場合は、各インスタンスのGPUの最大数と最小数を設定します。テスト目的では、1つのGPUに制限されます。本番稼働用ワークロードの追加をリクエストするには、[「Service Quotas ユーザーガイド」の「クォータの引き上げのリクエスト」](#)を参照してください。Service Quotas
12. フリートに必要な最小 vCPUsを入力します。
13. フリートに必要な最小メモリと最大メモリを入力します。
14. (オプション) フリートから特定のインスタンスタイプを許可または除外して、それらのインスタンスタイプのみがこのフリートに使用されるようにすることができます。

15. (オプション) フリートをスケーリングするインスタンスの最大数を設定し、キュー内のジョブで容量を使用可能にします。キューに入れられたジョブがないときにフリートがすべてのインスタンスを解放するように0、インスタンスの最小数を のままにしておくことをお勧めします。
16. [次へ] を選択します。
17. ストレージ機能で、フリートのストレージモードを選択します。

- 永続的ストレージ (推奨) – ワーカーライフサイクルイベント全体でキャッシュされたデータを保持し、アプリケーションキャッシュ、パッケージ、ワークスペースを維持することでコールドスタートの遅延を排除します。追加の Amazon Elastic Block Store (Amazon EBS) ストレージ料金が適用されます。
- ルートストレージのみ – ワーカーライフサイクルイベント間でデータをキャッシュしません。依存関係が最小限であるか、起動時間が短いジョブに最適です。

ブートボリュームのルートストレージ設定 (サイズ、IOPS、スループット) を設定します。永続ストレージを選択した場合は、永続ボリューム設定 (サイズ、マウントパス、スループット、最大アイドル時間、IOPS) も設定します。永続的ストレージの詳細については、「」を参照してください [サービスマネージドフリートの永続的ストレージ](#)。

18. [次へ] を選択します。
19. (オプション) このフリートの機能を定義するカスタムワーカー機能を定義し、ジョブの送信で指定されたカスタムホスト機能と組み合わせることができます。フリートを独自のライセンスサーバーに接続する場合は、特定のライセンスタイプがその一例です。
20. [次へ] を選択します。
21. (オプション) フリートをキューに関連付けるには、ドロップダウンからキューを選択します。キューがデフォルトのcondaキュー環境で設定されている場合、フリートにはパートナーの DCC アプリケーションとレンダラーをサポートするパッケージが自動的に提供されます。提供されているパッケージのリストについては、「」を参照してください [デフォルトのcondaキュー環境](#)。
22. [次へ] を選択します。
23. (オプション) フリートにタグを追加するには、新しいタグを追加を選択し、そのタグのキーと値を入力します。
24. [次へ] を選択します。
25. フリート設定を確認し、フリートの作成を選択します。

GPU アクセラレーターを使用する

1 つ以上の GPUs を使用してジョブの処理を高速化するように、サービスマネージドフリートでワーカーホストを設定できます。アクセラレーターを使用すると、ジョブの処理にかかる時間を短縮できますが、各ワーカーインスタンスのコストが増加する可能性があります。GPU アクセラレーターとそうでないフリートとのトレードオフを理解するには、ワークロードをテストする必要があります。

GPUs は、wait-and-save のインスタンスを持つフリートでは使用できません。

Note

テスト目的では、1 つの GPU に制限されます。本稼働ワークロードの詳細をリクエストするには、[「Service Quotas ユーザーガイド」の「クォータの引き上げのリクエスト」](#)を参照してください。Service Quotas

ワーカーインスタンス機能を指定するときに、フリートが GPU アクセラレーターを使用するかどうかを決定します。GPUs を使用する場合は、各インスタンスの GPUs の最小数と最大数、使用する GPU チップのタイプ、GPUs のランタイムドライバーを指定できます。

使用可能な GPU アクセラレーターは次のとおりです。

- T4 - NVIDIA T4 Tensor コア GPU
- A10G - NVIDIA A10G Tensor Core GPU
- L4 - NVIDIA L4 Tensor Core GPU
- L40s - NVIDIA L40S Tensor Core GPU
- RTX PRO Server 6000 - NVIDIA RTX PRO 6000 Blackwell Server Edition GPU

次のランタイムドライバーから選択できます。

- Latest - チップで利用できる最新のランタイムを使用します。を指定latestし、ランタイムの新しいバージョンがリリースされると、ランタイムの新しいバージョンが使用されます。
- grid:r580 - [NVIDIA vGPU ソフトウェア 19](#)。
- grid:r570 - [NVIDIA vGPU ソフトウェア 18](#)。
- grid:r535 - [NVIDIA vGPU ソフトウェア 16](#)。
- grid:r550 (非推奨) - [NVIDIA vGPU ソフトウェア 17](#)。

ランタイムを指定しない場合、Deadline Cloud はデフォルトlatestとして を使用します。ただし、複数のアクセラレーターがあり、一部のアクセラレーターlatestに を指定し、他のアクセラレーターを空白のままにすると、Deadline Cloud は例外を発生させます。

サービスマネージドフリートの永続的ストレージ

AWS Deadline Cloud (Deadline Cloud) 永続ストレージは、ルートブートボリュームとは別に、サービスマネージドフリート (SMF) ワーカー専用の Amazon Elastic Block Store (Amazon EBS) ボリュームを提供します。これらのボリュームは、ワーカーのライフサイクルイベント全体でデータを保持します。永続的ストレージでは、定期メンテナンスまたはスケーリングオペレーション中にワーカーが置き換えられた場合でも、conda パッケージのインストール、アプリケーションキャッシュ、アセットファイルは引き続き使用できます。

永続的ストレージの仕組み

サービスマネージドフリートで永続ストレージを有効にすると、Deadline Cloud はワーカーの Amazon EBS ボリュームのライフサイクルを自動的に管理します。

1. ワーカーが起動すると、Deadline Cloud は同じアベイラビリティーゾーンで使用可能な Amazon EBS ボリュームを作成または再利用し、そのボリュームをワーカーにアタッチします。
2. Deadline Cloud はボリュームをフォーマットし (新しい場合)、フリート設定で指定したパスにマウントします。
3. ワーカーが終了または置き換えられると、Deadline Cloud はボリュームをデタッチし、同じフリートとアベイラビリティーゾーンの将来のワーカーが再利用できるようにします。

ボリュームは同じフリートとアベイラビリティーゾーン内で再利用されるため、後続のワーカーは以前にボリュームに書き込まれたデータからメリットを得られます。ボリュームは、ワーカー間で競合することなく、専用帯域幅と IOPS を提供します。

Note

永続的ストレージは、サービスマネージドフリートでのみ使用できます。カスタマーマネージドフリートの場合は、独自のストレージインフラストラクチャを管理します。

永続的ストレージの利点

永続的ストレージは、サービスマネージドフリートワークロードに次の利点をもたらします。

- ジョブの起動の高速化 – Conda パッケージのインストール、コンパイルされたシェーダー、および処理されたアセットは、ワーカーの置き換え後も保持されるため、ダウンロードとインストールの繰り返しがなくなります。
- 専用パフォーマンス – 各ワーカーは、専用 IOPS とスループットを備えた独自の Amazon EBS ボリュームを受け取り、共有ネットワークストレージで発生する競合を回避します。
- 自動管理 – Deadline Cloud は、手動での介入を必要とせずに、ボリュームの作成、アタッチメント、フォーマット、マウント、クリーンアップを処理します。
- ランタイム統合 – conda キュー環境や仮想ファイルシステム (VFS) イミュータブルキャッシュなどのサポートされているランタイムコンシューマーは、使用可能な場合、ジョブ設定を変更することなく、自動的に永続的ストレージを使用します。
- コスト管理 – 未使用のボリュームを自動的にクリーンアップし、アイドル期間中のストレージコストを削減するように time-to-live (TTL) を設定します。

永続ストレージを使用するタイミング

以下のシナリオでは、サービスマネージドフリートの永続ストレージを有効にすることを検討してください。

- ジョブは、ダウンロードとインストールにかなりの時間がかかる conda パッケージを使用します。
- レンダリングワークロードはシェーダーをコンパイルするか、後続のレンダリングで再利用できるアセットを処理します。
- ワークスペースの同期状態によって後続の更新時のデータ転送が減少する Perforce またはその他のバージョン管理システムを使用します。
- ジョブは仮想ファイルシステム (VFS) を使用し、永続的なイミュータブルなアセットキャッシュからメリットを得られます。
- 共有ネットワークファイルシステムの管理に伴う運用上のオーバーヘッドなしで、専用のストレージパフォーマンスが欲しい。
- カスタムレンダラーやその他のソフトウェアをホスト設定スクリプトを介してワーカーにインストールし、それらのインストールを永続ボリュームにキャッシュするようにします。

フリートの永続ストレージの設定

永続ストレージは、新しいサービスマネージドフリートを作成するとき、または既存のフリートを更新するときに設定できます。

永続的ストレージの設定 (コンソール)

開始する前に、少なくとも 1 つのサービスマネージドフリートを持つ既存のファームがあるか、新しいフリートを作成する準備ができています。

フリートの永続ストレージを設定するには

1. AWS マネジメントコンソールにサインインし、[Deadline Cloud コンソール](#)を開きます。
2. ナビゲーションペインで、ファームを選択し、ファームを選択します。
3. フリートタブを選択し、フリートの作成を選択するか、既存のサービスマネージドフリートを選択して編集を選択します。
4. ストレージ機能で、ストレージモードで、永続ストレージを選択します。
5. ブートボリュームのルートストレージ設定 (サイズ、IOPS、スループット) を設定します。
6. 永続ストレージで、次の設定を行います。
 - サイズ – 永続ボリュームのサイズ。有効な範囲は 1~65,536 GiB です。デフォルトのサイズがレンダリングワークロードに適していることを確認し、より大きなアセットまたはキャッシュを使用するワークフローのボリュームサイズを増やすことを検討してください。
 - マウントパス – ボリュームがワーカーにマウントされる絶対パス (の場合など /mnt/persistentLinux)。Windows ワーカーの場合は、 などのドライブ文字を指定します D:。
 - スループット – ボリュームのプロビジョニングされたスループット。有効な範囲は 125~2,000 MiB/秒です。
 - 最大アイドル時間 – 使用可能なボリュームが削除されるまでにアイドル状態で待機できる時間。ドロップダウンから値を選択します (12 時間など)。
 - IOPS – ボリュームのプロビジョニングされた IOPS。有効な範囲は 3,000~80,000 IOPS です。IOPS は 4 倍以上のスループットである必要があります。
7. 残りのフリート設定ステップを完了し、フリートの作成または変更の保存を選択します。

永続ストレージの設定 (AWS CLI)

AWS Command Line Interface (AWS CLI) を使用して永続ストレージを設定するには、フリートを作成または更新するときにフリート設定に `persistentVolumeConfiguration` パラメータを含めます。

次の例では、永続ストレージが有効になっているフリートを作成します。

```
{
```

```
"configuration": {
  "serviceManagedEc2FleetConfiguration": {
    "instanceCapabilities": {
      "vCpuCount": {"min": 4, "max": 8},
      "memoryMiB": {"min": 16384, "max": 32768},
      "osFamily": "LINUX"
    },
    "instanceMarketOptions": {
      "type": "spot"
    },
    "persistentVolumeConfiguration": {
      "sizeGiB": 2048,
      "iops": 16000,
      "throughputMiB": 500,
      "mountPath": "/mnt/persistent",
      "lastUsedTtlHours": 168
    }
  }
}
```

ランタイム統合

永続的ストレージがワーカーに正常にマウントされると、Deadline Cloud は `DEADLINE_PERSISTENT_MOUNT` 環境変数を設定済みのマウントパスに設定します。次のランタイムコンシューマーは、環境変数が存在する場合、永続的ストレージを自動的に使用します。

- Conda キュー環境 – パッケージのインストールは永続ボリュームに保存されるため、後続のワーカーは、以前にインストールされたパッケージをダウンロードして再度インストールするのではなく、再利用します。
- 仮想ファイルシステム (VFS) イミュータブルキャッシュ – VFS はイミュータブルアセットキャッシュを永続ボリュームに保存するため、以前にダウンロードしたアセットは Amazon Simple Storage Service (Amazon S3) から再ダウンロードすることなく使用できます。

独自のジョブテンプレートとスクリプトで `DEADLINE_PERSISTENT_MOUNT` 環境変数を使用して、ワーカーライフサイクルイベント全体で保持されるデータを保存することもできます。

永続ボリュームの管理

Deadline Cloud コンソール、または API を使用して AWS CLI、フリートの永続ボリュームを表示および管理できます。次のオペレーションを使用できます。

- ボリュームを一覧表示する – 状態、サイズ、アベイラビリティゾーンなど、フリートに関連付けられているすべての永続ボリュームを表示します。
- ボリュームの詳細の取得 – 現在の状態、アタッチメントのステータス、設定など、特定のボリュームに関する詳細情報を取得します。
- ボリュームの削除 – 不要になったアタッチされていない永続ボリュームを完全に削除します。現在ワーカーにアタッチされているボリュームは削除できません。

永続的ストレージ設定の更新

既存のフリートの永続的ストレージ設定を更新できます。以下の変更がサポートされています。

- IOPS とスループット – Deadline Cloud は、次にボリュームをワーカーにアタッチするときに変更を適用します。
- ボリュームサイズ (増加のみ) – Deadline Cloud は、次のアタッチメントの前にボリュームを拡大します。ボリュームサイズを小さくすることはできません。
- 永続的ストレージを無効にする – Deadline Cloud は、既存のボリュームがワーカーにアタッチされなくなったときに、それらのボリュームを自動的にクリーンアップします。

Important

設定の変更は、既存のワーカーには影響しません。変更は、更新後に起動する新しいワーカーにのみ適用されます。

暗号化

永続ボリュームは、ファームレベルで設定された暗号化設定を使用します。ファームにカスタマーマネージド AWS Key Management Service (AWS KMS) キーを設定した場合、永続ボリュームはそのキーで暗号化されます。それ以外の場合、永続ボリュームはサービス所有のキーで暗号化されます。

考慮事項

永続ストレージを使用する場合は、次の考慮事項に注意してください。

- 永続的ボリュームはキャッシュの最適化であり、耐久性のあるプライマリストレージではありません。永続ボリュームは、パッケージのインストール、コンパイルされたシェーダー、アセット

キャッシュなど、再作成できるデータにのみ使用します。Deadline Cloud はいつでもボリュームを置き換えることができ、永続ボリュームに直接アクセスすることはできません。

- Deadline Cloud は、永続的ボリュームを使用するようにワーカーのホームディレクトリを設定します。ホームディレクトリにデータを保存するソフトウェア (conda パッケージやアプリケーションキャッシュなど) は、永続化によって自動的にメリットが得られます。ソフトウェアがホームディレクトリ外のパスに書き込む場合は、永続的なマウントパスを使用するように再設定する必要があります。そうしないと、これらのファイルはワーカーライフサイクルイベント全体で保持されません。
- 永続ボリュームは複数のワーカーに同時にアタッチされません。各ボリュームは一度に 1 人のワーカーを処理しますが、ライフサイクルイベント全体で異なるワーカーによって再利用されません。
- ボリュームは、特定のフリートとアベイラビリティゾーンに限定されます。1 つのアベイラビリティゾーンで作成されたボリュームを、別のアベイラビリティゾーンのワーカーが再利用することはできません。
- 特定のワーカーは、以前に使用したのと同じボリュームを受け取るとは限りません。同じフリートとアベイラビリティゾーンで使用可能なボリュームを割り当てることができます。
- 永続的ストレージをプロビジョニングできない場合 (クォータの制限など)、ジョブは失敗します。ワーカーは、永続的ストレージなしでは実行にフォールバックしません。
- 永続ストレージには、アクティブなボリュームの数とその設定に基づいて課金されます。アイドル期間中のコストを制御するには、TTL を設定するか、フリートから永続的ストレージ設定を削除します。

サービスマネージドフリートのソフトウェアライセンス

Deadline Cloud は、一般的に使用されるソフトウェアパッケージの使用ベースのライセンス (UBL) を提供します。サポートされているソフトウェアパッケージは、サービスマネージドフリートで実行されると、自動的にライセンスされます。ソフトウェアライセンスサーバーを設定または保守する必要はありません。ライセンスはスケーリングされるため、大規模なジョブでは使い果たされません。

組み込みの Deadline Cloud conda チャンネルを使用して UBL をサポートするソフトウェアパッケージをインストールするか、独自のパッケージを使用できます。conda チャンネルの詳細については、「」を参照してください [キュー環境を作成する](#)。

サポートされているソフトウェアパッケージのリストと UBL の料金については、[AWS 「Deadline Cloud の料金」](#) を参照してください。

サービスマネージドフリートで独自のライセンスを使用する

Deadline Cloud 使用量ベースのライセンス (UBL) では、ソフトウェアベンダーとの個別のライセンス契約を管理する必要はありません。ただし、既存のライセンスがある場合、または UBL で利用できないソフトウェアを使用する必要がある場合は、Deadline Cloud サービスマネージドフリートで独自のソフトウェアライセンスを使用できます。インターネット経由で SMF をソフトウェアライセンスサーバーに接続して、フリート内の各ワーカーのライセンスをチェックアウトします。

プロキシを使用してライセンスサーバーに接続する例については、「Deadline Cloud Developer Guide」の「[Connect service-managed fleets to a custom license server](#)」を参照してください。

VFX Reference Platform の互換性

VFX Reference Platform は VFX 業界共通のターゲットプラットフォームです。をサポートするソフトウェアで Amazon Linux 2023 を実行する標準サービスマネージドフリート Amazon EC2 インスタンスを使用するには VFX Reference Platform、サービスマネージドフリートを使用するとき以下の考慮事項に留意する必要があります。

VFX Reference Platform は毎年更新されます。Deadline Cloud サービスマネージドフリートを含む AL2023 を使用する際のこれらの考慮事項は、2022 年から 2024 年までの暦年 (CY) リファレンスプラットフォームに基づいています。詳細については、「[VFX Reference Platform](#)」を参照してください。

Note

カスタマーマネージドフリートのカスタム Amazon Machine Image (AMI) を作成する場合は、Amazon EC2 インスタンスを準備するときにこれらの要件を追加できます。

AL2023 Amazon EC2 インスタンスで VFX Reference Platform サポートされているソフトウェアを使用するには、次の点を考慮してください。

- AL2023 と共にインストールされる glibc バージョンはランタイム用に互換性がありますが、CY2024 VFX Reference Platform 以前と互換性のあるソフトウェアの構築には互換性がありません。
- Python 3.9 および 3.11 にはサービスマネージドフリートが用意されており、CY2022 および VFX Reference Platform CY2024 と互換性があります。Python 3.7 および 3.10 は、サービスマネージドフリートでは提供されません。それらを必要とするソフトウェアは、キューまたはジョブ環境に Python インストールを提供する必要があります。

- サービスマネージドフリートで提供される一部の Boost ライブラリコンポーネントはバージョン 1.75 であり、と互換性がありません VFX Reference Platform。アプリケーションが Boost を使用している場合は、互換性のためにライブラリの独自のバージョンを指定する必要があります。
- Intel TBB 更新 3 は、サービスマネージドフリートで提供されます。このバージョンは VFX Reference Platform、CY2022、CY2023、および CY2024 と互換性があります。
- で指定されたバージョンを持つ他のライブラリ VFX Reference Platform は、サービスマネージドフリートによって提供されません。サービスマネージドフリートで使用されるすべてのアプリケーションをライブラリに提供する必要があります。ライブラリのリストについては、「[リファレンスプラットフォーム](#)」を参照してください。

ワーカー AMI ソフトウェアの内容

このセクションでは、Deadline Cloud AWS のサービスマネージドワーカー (AMIs) にインストール Amazon Machine Image されるソフトウェアについて説明します。

AWS Deadline Cloud のサービスマネージドワーカー AMIs は、Windows Server 2022 と Amazon Linux 2023 の両方に基いており、レンダリングワークロードをサポートするために特別にインストールされた追加のソフトウェアが含まれています。これらの AMIs は、機能を維持するために継続的に更新されます。

これらの AMIs は、次のいずれかのサポートカテゴリに分類されます。

サービスが提供するソフトウェアパッケージ

ワークロードのレンダリング用に特別にインストールおよび保守されているソフトウェア
追加のシステムソフトウェア

予告なしに変更される可能性のある他のすべてのソフトウェア

サービスが提供するソフトウェアパッケージ

これらのソフトウェアパッケージは、レンダリングワークロードをサポートするためにインストールされ、互換性のために維持されます。これらのパッケージへの依存関係を安全に取得できます。

開発ツールと言語

Linux (AL2023):

- Python 3.11

- Git

Windows (Server 2022):

- Python 3.11
- の Git Windows

AWS ツール

両方のプラットフォーム:

- AWS コマンドラインインターフェイス v2 (AWS CLI v2)

システムライブラリとユーティリティ

Linux:

- ファイルシステムオペレーション用の FUSE および FUSE3 ライブラリ
- イメージライブラリ
 - libpng
 - libjpeg
 - libtiff
- OpenGL ライブラリ
 - mesa-libGLU
 - mesa-libGL
 - mesa-libEGL
 - libglvnd-opengl
- 開発ライブラリ:
 - json-c (JSON 解析)
 - libnsl (ネットワークサービスライブラリ)
 - libxcrypt-compat (暗号化互換性)
- X ウィンドウライブラリ
 - libXmu
 - libXpm

- libXinerama
- libXcomposite
- libXrender
- libXrandr
- libXcursor
- libXi
- libxdamage
- libXtst
- libxkbcommon
- libSM
- ネットワークおよびシステムユーティリティ
 - tcsh

GPU アクセラレーションフリート

- Nvidia グリッドドライバー

パッケージマネージャー

Linux:

- conda/Mamba パッケージマネージャー (にインストール/opt/conda)
- DNF パッケージマネージャー (システムパッケージ)
- pip (Python パッケージインストーラ)

Windows:

- conda/Mamba パッケージマネージャー (にインストールC:\ProgramData\conda)
- pip (Python パッケージインストーラ)

追加のシステムソフトウェア

AMI 上の他のすべてのソフトウェアは、予告なしに更新、削除、または変更できます。上記の「サポートされているソフトウェアパッケージ」セクションに明示的に記載されていないソフトウェアには依存しないでください。この制限には以下が含まれますが、これらに限定されません。

- オペレーティングシステムパッケージとライブラリ
- サービス管理コンポーネント
- 基本 AMI ソフトウェアとドライバー
- ソフトウェアの依存関係とランタイムライブラリ
- システム設定ツールとユーティリティ

その他のシステムソフトウェアの例

Linux: systemd、カーネルモジュール、ハードウェアドライバー、ネットワークコンポーネント、およびベース AL2023 ディストリビューションの一部としてインストールされるサポートライブラリなどのシステムパッケージ。

Windows: Windowsシステムコンポーネント、Microsoft Edge、Amazon EC2 サービスソフトウェア、ハードウェアドライバー、Windowsランタイムコンポーネント。

ベストプラクティス

依存関係管理: サポートされているソフトウェアパッケージセクションに記載されているソフトウェアにのみ依存します。

パッケージバージョン: 特定のソフトウェアバージョンでは、AMI が提供するバージョンに依存するのではなく、パッケージマネージャー (pip、conda など) を使用して特定のパッケージをインストールします。

環境の分離: 仮想環境 (Python venv や conda 環境など) を使用して、特定の依存関係を分離します。

AMI 更新モデル

ワーカー AMI の更新方法に関する以下の情報に注意してください。

- ワーカー AMIs はバージョンニングシステムなしで継続的に更新されます。
- 更新は、サービスオペレーションの一部として自動的に行われます。
- AMI 更新には事前通知システムは提供されません。

カスタマーマネージドフリート

管理するワーカーのフリートを使用する場合は、Deadline Cloud がジョブの処理に使用するカスタマーマネージドフリート (CMF) を作成できます。次の場合は CMF を使用します。

- Deadline Cloud と統合する既存のオンプレミスワーカーがあります。
- 同じ場所にあるデータセンターにワーカーがいる。
- Amazon Elastic Compute Cloud (Amazon EC2) ワーカーを直接制御したい。

CMF を使用すると、フリートを完全に制御し、その責任を担います。これには、フリート内のワーカーのプロビジョニング、運用、管理、廃止が含まれます。

詳細については、[「Deadline Cloud デベロッパーガイド」の「Deadline Cloud カスタマーマネージドフリートの作成と使用」](#)を参照してください。

自動スケーリング設定

Deadline Cloud には、フリートによるワーカーのスケールアップとスケールダウンの方法をカスタマイズできる自動スケーリング設定オプションが用意されています。これらの設定は、ワークフロー要件に基づいてジョブ処理速度とコスト効率のバランスを取るのに役立ちます。

フリートには、次の自動スケーリング設定を設定できます。

- 最小ワーカー数 – フリートで常に維持されているワーカーの最小数を指定します。
- 最大ワーカー数 – 同時に実行できるワーカーの数を制限します。
- スケールアウトレート – ワーカーをフリートに追加する速度を制御します。
- ワーカーのアイドル時間 – ワーカーがシャットダウンする前に新しい作業を待機する時間を制御します。
- スタンバイワーカー数 – アイドル状態のワーカーのウォームスタンバイプールを維持し、ジョブをすばやく開始します。

自動スケーリングの仕組みは、フリートタイプによって異なります。

- サービスマネージドフリート – Deadline Cloud は、設定に基づいて自動スケーリングを自動的に実装します。設定を構成すると、サービスはワーカーのプロビジョニングを処理します。

- カスタマーマネージドフリート – カスタマーマネージドフリートの自動スケーリング設定を完了している場合、自動スケーリング設定はサービスマネージドフリートの場合と同じように機能します。サービスは設定を使用して希望する容量を計算し、推奨されるフリートサイズのイベントをフリートに送信します。詳細については、「[Deadline Cloud Developer Guide](#)」の「[Set up auto scaling for customer-managed fleets](#)」を参照してください。

スケールアウトレート

スケールアウトレート (scaleOutWorkersPerMinute) 設定は、フリートがスケールアウトしたときに 1 分あたりに起動を開始するワーカーの数を制御します。Amazon EC2 インスタンスの起動には数分かかる場合があるため、ワーカーはすぐには利用できない場合があります。

スケールアウトレートを設定するときは、次の点を考慮してください。

- レートが高いほど、より多くのワーカーをすばやく起動できるため、大規模なジョブのジョブ完了時間を短縮できます。
- レートが高いと、存続期間の短いタスクに必要な数よりも多くのワーカーを起動し、コストが増加する可能性があります。
- レートを低くすると、ジョブの失敗を早期に検出し、失敗したジョブの無駄なコンピューティングによるコストを削減できます。
- 存続期間の短いタスクの場合、ワーカーが実際のタスク実行よりも環境のロードに費やす時間が短いため、保守的なスケーリングアプローチの方が費用対効果が高くなります。

Note

スケールアウトレートはベストエフォート設定です。実際のスケーリング速度は、インスタンスの可用性やその他のシステム要因によって異なる場合があります。まれに、実際のレートが設定値を一時的に超えることがあります。

ワーカーのアイドル時間

ワーカーアイドル時間 (workerIdleDurationSeconds) 設定は、ジョブの処理終了後にワーカーが使用可能な状態を維持する期間を秒単位で制御します。デフォルト値は 300 秒 (5 分) です。

この設定は、アーティストが頻繁にジョブを修正して再送信する反復ワークフローに役立ちます。ワーカーの可用性を長く維持することで、後続のジョブ送信は、新しいワーカーの起動を待たずにすぐに処理を開始できます。

ワーカーのアイドル時間を設定するときは、次の点を考慮してください。

- 期間が長くなると、ワーカーは迅速に反復できるため、ジョブの送信間の待機時間が短縮されます。ただし、アイドル状態のワーカーには引き続き料金が発生するため、期間が長くなるとコストが増加します。
- 期間が短いほど、アイドル状態のワーカーをより迅速にシャットダウンできるため、コストを削減できます。
- サービスマネージドフリートの場合、ワーカーは 24 時間ごとに更新されるため、最大値は 86,400 秒 (24 時間) です。ワーカーが 23 時間実行されており、アイドル時間を 10 時間に設定した場合、ワーカーは 24 時間の制限に達すると 1 時間後にシャットダウンします。

スタンバイワーカー数

スタンバイワーカー数 (standbyWorkerCount) 設定は、ウォームスタンバイプールとして維持するアイドルワーカーの数を指定します。これらのワーカーは、新しいインスタンスの起動を遅らせることなく、新しいジョブを処理できます。

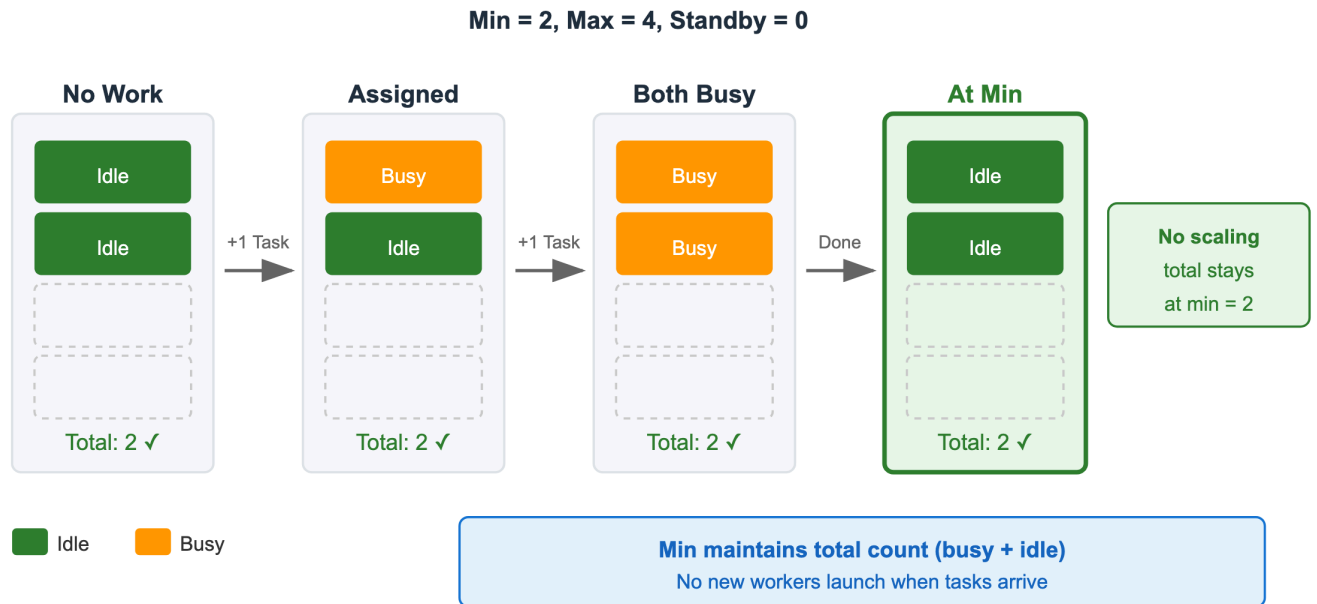
この設定は、ジョブ開始のレイテンシーを短縮する場合に便利です。たとえば、スタンバイワーカーは、Windows インスタンスでレンダリングする場合、ローカル依存関係をインストールするホスト設定スクリプトを使用する場合、またはワーカーがかなりのセットアップ時間を必要とする場合に役立ちます。フリートは設定された数のアイドルワーカーを維持しようとしませんが、代替ワーカーの起動中にアイドル数が一時的に減少することがあります。

スタンバイワーカー数を設定するときは、次の点を考慮してください。

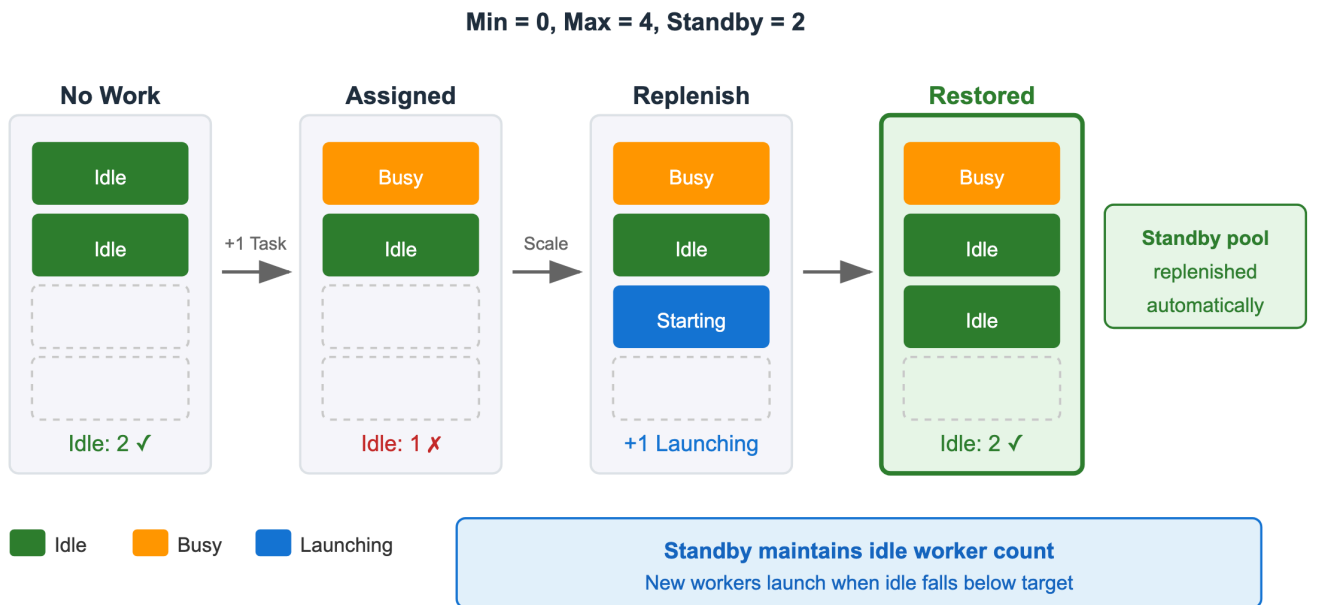
- スタンバイワーカーは、ジョブを処理していない場合でもコストが発生します。スタンバイワーカーの数を予算とジョブ開始のレイテンシー要件と照らし合わせます。
- フリートがワーカーの最大数に達すると、スタンバイプールが完全に維持されない可能性があります。たとえば、すべてのワーカーがビジー状態で、フリートが最大サイズである場合、追加のアイドルワーカーは起動されません。
- スタンバイワーカー数が最小ワーカー数を超えると、最小ワーカー数は実質的に上書きされます。たとえば、最小 1、スタンバイ 2 の場合、フリートは作業が利用できないときに 2 人のアイドルワーカーを保持し、最小設定を冗長にします。

次の図は、最小ワーカー数とスタンバイワーカー数がフリートスケーリング動作にどのように影響するかを示しています。タブを選択すると、各シナリオが表示されます。

Minimum worker count



Standby worker count



スケジュールに基づいてスタンバイワーカーの数を自動的に調整するには、GitHub の [fleet_standby_scheduling](#) で sample AWS CloudFormation (CloudFormation) テンプレートを使用します。

自動スケーリング設定の構成

自動スケーリング設定は、フリートの作成時または既存のフリートの更新時に設定できます。

自動スケーリング設定を構成するには

1. [Deadline Cloud コンソール](#)を開きます。
2. フリートを含むファームに移動します。
3. フリートタブを選択します。
4. 設定するフリートを選択し、編集を選択します。
5. Auto Scaling セクションで、次の設定を行います。
 - 最小ワーカー数 – 維持するワーカーの最小数を入力します。
 - 最大ワーカー数 – 許可されるワーカーの最大数を入力します。
 - スケールアウトレート – 1分あたりに起動するワーカーの数を入力します。
 - ワーカーのアイドル時間 – シャットダウンするまでワーカーがアイドル状態のままになる秒数を入力します。
 - スタンバイワーカー数 – 維持するスタンバイワーカーの数を入力します。
6. [Save changes] (変更の保存) をクリックします。

Deadline Cloud でのユーザーの管理

AWS Deadline Cloud は AWS IAM アイデンティティセンター を使用してユーザーとグループを管理します。IAM Identity Center は、エンタープライズシングルサインオン (SSO) プロバイダーと統合できるクラウドベースのシングルサインオンサービスです。統合により、ユーザーは会社のアカウントでサインインできます。

Deadline Cloud はデフォルトで IAM Identity Center を有効にし、Deadline Cloud をセットアップして使用する必要があります。の組織所有者 AWS Organizations は、Deadline Cloud モニターにアクセスできるユーザーとグループを管理する責任があります。詳細については、「[とは AWS Organizations](#)」を参照してください。

ユーザーを管理する方法は、IAM Identity Center の ID ソース設定によって異なります。ID ソースは、IAM Identity Center がユーザー情報を取得する場所を定義します。

トピック

- [ID ソースについて](#)
- [を使用したユーザーの作成と管理 IAM アイデンティティセンターディレクトリ](#)
- [外部 ID プロバイダーを使用してユーザーを管理する](#)
- [モニターにアクセスできるユーザーの制限](#)
- [アクセスレベルについて](#)

ID ソースについて

IAM Identity Center は ID ソースを使用して、ユーザーの管理場所を定義します。ID ソースには 2 つのタイプがあります。

IAM アイデンティティセンターディレクトリ

これはデフォルトの ID ソースです。ユーザーは IAM Identity Center 内で直接作成および管理されます。ユーザーを作成するには、Deadline Cloud コンソールまたは IAM Identity Center コンソールを使用します。ユーザーは組織に参加するための招待メールを受け取り、パスワードは IAM アイデンティティセンター内で管理されます。

外部 ID プロバイダー (IdP)

ユーザーは、Okta、Microsoft Entra IDまたはその他の SAML 2.0 ID プロバイダーなどの外部システムからフェデレーションされます。ユーザーはまず外部システムで作成する必要があります。

す。外部 IdP が設定されている場合、Deadline Cloud コンソールはユーザーを作成できませんが、既存のユーザーにアクセス許可を割り当てることができます。パスワードは外部 IdP によって管理されます。

ID ソースの設定を確認または変更するには、IAM Identity Center ユーザーガイドの「[ID ソースの管理](#)」を参照してください。

を使用したユーザーの作成と管理 IAM アイデンティティセンターディレクトリ

ID ソースが に設定されている場合 IAM アイデンティティセンターディレクトリ、Deadline Cloud コンソールから直接ユーザーとグループを作成および管理できます。コンソールで作成されたユーザーは、IAM Identity Center から E メール招待を受け取ります。招待を承諾すると、ユーザーは Deadline Cloud モニターにアクセスできます。

Note

IAM アイデンティティセンターが外部 ID プロバイダーに接続されている場合、Deadline Cloud コンソールを使用してユーザーを作成することはできません。外部 [the section called “外部 IdP を使用してユーザーを管理する”](#) IdP でユーザーを管理する方法については、「」を参照してください。

1. にサインイン AWS マネジメントコンソール し、Deadline Cloud [コンソール](#)を開きます。メインページの「開始方法」セクションで、「Deadline Cloud のセットアップ」または「ダッシュボードに移動」を選択します。
2. 左側のナビゲーションペインで、ユーザー管理を選択します。デフォルトでは、グループタブが選択されています。

実行するアクションに応じて、グループタブまたはユーザータブを選択します。

Groups


グループを作成するには

1. [グループの作成] を選択してください。

2. グループ名を入力します。名前は、IAM Identity Center 組織内のグループ間で一意である必要があります。

グループを削除するには

1. 削除するグループを選択します。
2. [を削除] を選択します。
3. 確認ダイアログで、グループの削除を選択します。

 Note

IAM Identity Center からグループを削除します。グループメンバーは、Deadline Cloud にサインインしたり、ファームリソースにアクセスしたりできなくなります。

Users

ユーザーを追加するには

1. [ユーザー] タブを選択します。
2. [ユーザーの追加] を選択します。
3. 新しいユーザーの名前、E メールアドレス、ユーザー名を入力します。
4. (オプション) 新しいユーザーを追加する 1 つ以上の IAM Identity Center グループを選択します。
5. 招待を送信を選択して、IAM Identity Center 組織に参加する手順が記載された E メールを新しいユーザーに送信します。

ユーザーを削除するには、次の手順を実行します

1. 削除するユーザーを選択します。
2. [を削除] を選択します。
3. 確認ダイアログで、ユーザーの削除を選択します。

Note

IAM Identity Center からユーザーを削除します。ユーザーは Deadline Cloud Monitor にサインインしたり、ファームリソースにアクセスしたりできなくなります。

外部 ID プロバイダーを使用してユーザーを管理する

IAM アイデンティティセンターが Okta や などの外部 ID プロバイダー (IdP) に接続されている場合は Microsoft Entra ID、その外部システムでユーザーを作成して管理する必要があります。外部 IdP が設定されている場合、Deadline Cloud コンソールは新しいユーザーを作成できません。

ユーザーが外部 IdP で作成され、IAM Identity Center に同期されたら、Deadline Cloud リソースにアクセス許可を割り当てることができます。ファーム、キュー、フリートレベルでのアクセス許可の割り当てについては、[the section called “アクセスレベルについて”](#)「」を参照してください。

外部 ID プロバイダー設定の管理については、IAM Identity Center ユーザーガイドの「[ID ソースの管理](#)」を参照してください。

モニターにアクセスできるユーザーの制限

Deadline Cloud モニターにサインインするには、ユーザーに 2 つのレベルのアクセスが必要です。

- AWS IAM アイデンティティセンター (IAM Identity Center) アプリケーションアクセス — Deadline Cloud モニターアプリケーションにサインインするためのアクセス許可。
- Deadline Cloud リソースアクセス – サインイン後にファーム、キュー、その他のリソースを表示するアクセス許可。

デフォルトでは、IAM Identity Center アプリケーションへのアクセスは、ID ストア内のすべてのユーザーに開かれています。Deadline Cloud リソースアクセスは、プライマリアクセスコントロールレイヤーです。ただし、モニターアプリケーションにサインインできるユーザーを制限することで、2 番目の制御レイヤーを追加できます。

モニターのサインインを制限するには、IAM アイデンティティセンターの Deadline Cloud モニターアプリケーションで割り当てを要求する設定を有効にします。この設定を有効にすると、アプリケーションに明示的に割り当てたユーザーとグループのみがモニターにサインインできます。

モニターアプリケーションの必須割り当てを有効にするには

1. <https://console.aws.amazon.com/singlesignon> で IAM Identity Center コンソールを開きます。
2. ナビゲーションペインで、[アプリケーション] を選択します。
3. Deadline Cloud Monitor アプリケーションを選択します。
4. アプリケーションの割り当て設定で、必須の割り当てを有効にします。
5. アプリケーションにアクセスする必要がある特定のユーザーとグループを割り当てます。

アプリケーションに割り当てないユーザーやグループは、IAM Identity Center ID ストアに存在していても、モニターにサインインできません。

アプリケーションの割り当ての詳細については、IAM Identity Center ユーザーガイドの「[アプリケーションへのユーザーアクセスの割り当て](#)」を参照してください。

アクセスレベルについて

ID ソースに関係なく、Deadline Cloud コンソールを使用して、ファーム、キュー、フリートレベルのユーザーとグループに許可を割り当てます。アクセス許可は、さまざまなレベルで付与できます。後続の各レベルには、前のレベルのアクセス許可が含まれます。次のリストでは、最低レベルから最高レベルまでの 4 つのアクセスレベルについて説明します。

- ビューワー – アクセスできるファーム、キュー、フリート、ジョブ内のリソースを表示するアクセス許可。ビューワーはジョブを送信または変更できません。
- 寄稿者 – ビューワーと同じですが、キューまたはファームにジョブを送信するアクセス許可があります。
- マネージャー – 寄稿者と同じですが、アクセスできるキュー内のジョブを編集し、アクセスできるリソースに対するアクセス許可を付与するアクセス許可があります。
- 所有者 – マネージャーと同じですが、予算を表示および作成し、使用状況を確認できます。

これらのアクセスレベルのカスタマイズについては、「Deadline Cloud Developer Guide」の「[Monitor role](#)」を参照してください。

トピック

- [アクセスレベルのアクセス許可マトリックス](#)
- [メンバーシップの継承](#)

- [ユーザーとグループにアクセス許可を割り当てる](#)

アクセスレベルのアクセス許可マトリックス

次の表は、デフォルトの AWS マネージドポリシーを使用する場合に、ファーム、キュー、フリートの各アクセスレベルで利用できる特定のアクセス許可を示しています。現在、ユーザーアクセスの管理は Deadline Cloud コンソールでのみ使用でき、Deadline Cloud モニターでは利用できません。これらのアクセスレベルのカスタマイズについては、「Deadline Cloud Developer Guide」の「[Monitor role](#)」を参照してください。

アクセスレベル別のファームアクセス許可

アクセス許可	ビューワー	コントロール ビューター	Manager	所有者
ファームの詳細を表示する	はい	はい	はい	はい
キューとフリートを表示する	はい	はい	はい	はい
ジョブの送信	いいえ	はい	はい	はい
ユーザーアクセスの管理	いいえ	なし	はい	はい
予算の表示と作成	いいえ	いいえ	なし	はい
使用状況データの表示	いいえ	いいえ	なし	はい

アクセスレベル別のキューのアクセス許可

アクセス許可	ビューワー	コントロール ビューター	Manager	所有者
キューの詳細を表示する	はい	はい	はい	はい
キューでジョブを表示する	はい	はい	はい	はい
キューにジョブを送信する	いいえ	はい	はい	はい
ジョブの編集とキャンセル	いいえ	なし	はい	はい

アクセス許可	ビューワー	コントロ ビューター	Manager	所有者
キューユーザーアクセスの管理	いいえ	なし	はい	はい
キューの予算割り当てを表示する	いいえ	いいえ	なし	はい

アクセスレベル別のフリートのアクセス許可

アクセス許可	ビューワー	コントロ ビューター	Manager	所有者
フリートの詳細の表示	はい	はい	はい	はい
フリートのワーカーを表示する	はい	はい	はい	はい
フリートユーザーアクセスの管理	いいえ	なし	はい	はい
フリートのコストデータを表示する	いいえ	いいえ	なし	はい

メンバーシップの継承

Deadline Cloud は、ファーム、キュー、またはフリートレベルでアクセス許可を割り当てることのできる階層メンバーシップモデルを使用します。メンバーシップ継承の仕組みを理解すると、アクセスコントロールを効果的に設定できます。

ファームレベルのメンバーシップ

ファームレベルでユーザーまたはグループのメンバーシップを割り当てると、そのメンバーシップはファーム内のすべてのキューとフリートに適用されます。ファームレベルのメンバーシップは幅広いアクセスを提供し、複数のキューまたはフリートで作業する必要があるユーザーに役立ちます。

たとえば、ファームレベルでコントリビューターとしてユーザーを割り当てると、そのユーザーはファーム内の任意のキューにジョブを送信できます。

キューおよびフリートレベルのメンバーシップ

キューまたはフリートレベルでメンバーシップを割り当てて、より詳細なアクセスコントロールを行うこともできます。キューレベルおよびフリートレベルのメンバーシップは、その特定のリソースにのみ適用されます。

たとえば、特定のキューでユーザーをマネージャーとして割り当てた場合、そのユーザーはジョブを編集し、ファーム内の他のキューではなく、そのキューに対してのみアクセスを管理できます。

ユーザーは、ファームレベルのメンバーシップを持たないキューまたはフリートにのみアクセスできます。この場合、ユーザーはファームリストにファームを表示することはできませんが、ジョブを送信し、アクセスできるキューまたはフリートのみを表示できます。

有効なアクセス権限

ユーザーが複数のレベルでメンバーシップを持っている場合、Deadline Cloud は最も高いアクセスレベルを使用します。例えば、次のようになります。

- ファームレベルでビューワアクセスと特定のキューに対するマネージャーアクセスを持つユーザーには、そのキューに対するマネージャーアクセス許可と、他のすべてのキューに対するビューワアクセス許可があります。
- ファームレベルで Contributor アクセスがあり、特定のフリートの所有者アクセスを持つユーザーには、そのフリートに対する所有者アクセス許可と、他の場所での Contributor アクセス許可があります。

Note

ファーム、キュー、またはフリートレベルでメンバーシップを持たないユーザーは、IAM Identity Center を介して認証されていても、それらのリソースにアクセスできません。

ユーザーとグループにメンバーシップを割り当てる手順については、「」を参照してください [the section called “アクセス許可の割り当て”](#)。

ユーザーとグループにアクセス許可を割り当てる

Deadline Cloud コンソールを使用して、ファーム、キュー、またはフリートレベルのユーザーとグループにアクセスレベルを割り当てます。

Note

アクセス許可の変更がシステムに反映されるまでに最大 10 分かかる場合があります。

アクセス管理に移動するには

1. にサインイン AWS マネジメントコンソール し、Deadline Cloud [コンソール](#)を開きます。
2. 左側のナビゲーションペインで、ファームやその他のリソースを選択します。
3. 管理するファームを選択します。ファーム名を選択して詳細ページを開きます。検索バーを使用してファームを検索できます。
4. (オプション) ファームの代わりにキューまたはフリートを管理するには、キューまたはフリートタブを選択し、管理するキューまたはフリートを選択します。
5. アクセス管理タブを選択します。

実行するアクションに応じて、グループタブまたはユーザータブを選択します。

Groups

グループを追加するには

1. グループトグルを選択します。
2. [グループの追加] を選択します。
3. ドロップダウンから、追加するグループを選択します。
4. グループアクセスレベルで、次のいずれかのオプションを選択します。
 - 表示者
 - 寄稿者
 - Manager
 - [所有者]
5. [Add] (追加) を選択します。

グループを削除するには

1. 削除するグループを選択します。
2. [を削除] を選択します。
3. 確認ダイアログで、グループの削除を選択します。

Users

ユーザーを追加するには

1. ユーザーを追加するには、ユーザーの追加を選択します。
2. ドロップダウンから、追加するユーザーを選択します。
3. ユーザーアクセスレベルで、次のいずれかのオプションを選択します。
 - 表示者
 - 寄稿者
 - Manager
 - [所有者]
4. [Add] (追加) を選択します。

ユーザーを削除するには

1. 削除するユーザーを選択します。
2. [を削除] を選択します。
3. 確認ダイアログで、ユーザーの削除を選択します。

Deadline Cloud ジョブ

ジョブは、Deadline Cloud AWS が使用可能なワーカーの作業をスケジュールして実行するために使用する一連の手順です。ジョブを作成するときは、ジョブを送信するファームとキューを選択します。

送信者は、デジタルコンテンツ作成 (DCC) アプリケーションのプラグインであり、DCC アプリケーションのインターフェイスでのジョブの作成を管理します。ジョブを作成したら、送信者を使用して Deadline Cloud に送信し、処理します。

送信者は、[ジョブを記述する Open Job Specification \(OpenJD\)](#) テンプレートを作成します。同時に、アセットファイルを Amazon Simple Storage Service (Amazon S3) バケットにアップロードします。アップロード時間を短縮するために、送信者は Amazon S3 への前回のアップロード以降に変更されたファイルのみを送信します。

次の方法でジョブを作成することもできます。

- ターミナルから – コマンドラインを使用できるジョブを送信するユーザー向け。
- スクリプトから – ワークロードをカスタマイズおよび自動化します。
- アプリケーションから – ユーザーの作業がアプリケーションにある場合、またはアプリケーションのコンテキストが重要である場合。

詳細については、[Deadline Cloud デベロッパーガイドの「Deadline Cloud にジョブを送信する方法」](#)を参照してください。

ジョブは以下で構成されます。

- Priority – Deadline Cloud がキュー内のジョブを処理するおおよその順序。ジョブの優先度は 0 ~ 100 の間で設定できます。優先度の高いジョブは通常、最初に処理されます。優先度が同じジョブは、受信した順序で処理されます。
- ステップ – ワーカーで実行するスクリプトを定義します。ステップには、ワーカーの最小メモリや、最初に完了する必要があるその他のステップなどの要件があります。各ステップには 1 つ以上のタスクがあります。
- タスク – ワーカーに送信される作業単位。タスクは、ステップのスクリプトと、スクリプトで使用されるフレーム番号などのパラメータの組み合わせです。ジョブは、すべてのステップのすべてのタスクが完了すると完了します。

- 環境 – 複数のステップまたはタスクで共有される指示を設定および削除します。

Deadline Cloud 送信者の使用

送信者は、レンダリングジョブを Deadline Cloud に直接送信できるように、デジタルコンテンツの作成と統合するツールです。この統合により、アプリケーション間の切り替えやファイルを手動で転送する必要がなくなるため、ワークフローが効率化されます。これにより、時間が節約され、エラーが発生する可能性が低くなります。

送信者は、多くの一般的な DCC アプリケーションで使用できます。送信者をインストールすると、は通常、レンダリング設定またはエクスポートメニューで、Deadline Cloud 固有のオプションをアプリケーションのインターフェイスに追加します。

Deadline Cloud 送信者を使用すると、次のことができます。

- 使い慣れた DCC 環境でレンダージョブパラメータを設定する
- アプリケーションを離れずに Deadline Cloud にジョブを送信する
- 手動ファイル転送に関連するエラーの可能性を減らす
- アプリケーションを切り替える必要がないため、時間を節約できます。

DCC アプリケーションの送信者を検索するには、[ワークステーションのセットアップ](#)ページを確認してください。次に、[ワークステーションのセットアップ](#)「」の手順に従って送信者をインストールします。

アプリケーションにサポートされている送信者がいない場合でも、アプリケーションのジョブを実行できます。サンプルジョブバンドルが使用可能な場合もあれば、アプリケーションの render CLI コマンド用のシンプルな送信者を構築することもできます。詳細については、[「Deadline Cloud デベロッパーガイド」の「Deadline Cloud の Open Job Description \(OpenJD\) テンプレート](#)」を参照してください。

このトピックの例ではBlender送信者を使用していますが、他の送信者を使用する手順は似ています。

Note

送信者を使用するには、Deadline Cloud モニターにサインインする必要があります。

送信者には 4 つのタブがあります。

トピック

- [共有ジョブ設定タブ](#)
- [ジョブ固有の設定タブ](#)
- [ジョブアタッチメントタブ](#)
- [ホスト要件タブ](#)

共有ジョブ設定タブ

The screenshot shows a window titled "Submit to AWS Deadline Cloud" with four tabs: "Shared job settings", "Job-specific settings", "Job attachments", and "Host requirements". The "Shared job settings" tab is active and contains the following sections:

- Job Properties**
 - Name: testCube
 - Description: (empty)
 - Priority: 50
 - Initial state: READY
 - Maximum failed tasks count: 20
 - Maximum retries per task: 5
 - Maximum worker count: No max worker count, Set max worker count
- Deadline Cloud settings**
 - Farm: DocTestMonitor farm
 - Queue: DocTestMonitor queue
- Queue Environment: Conda**
 - Conda Packages: blender=4.2.* blender-openjd=0.5.*
 - Conda Channels: deadline-cloud

At the bottom, there are three status boxes: "Credential source" (DEADLINE_CLOUD_MONITOR_LOGIN), "Authentication status" (AUTHENTICATED), and "AWS Deadline Cloud API" (AUTHORIZED). Below these are buttons for "Login", "Logout", "Settings...", "Submit", and "Export bundle".

共有ジョブ設定タブには、送信者を使用して Deadline Cloud に送信されるすべてのジョブに共通の設定が含まれています。3つのセクションは次のとおりです。

- ジョブプロパティ – ジョブの全体的なプロパティを設定します。これらのプロパティは、すべての DCC アプリケーションの送信者に存在します。
- Deadline Cloud 設定 – ジョブが送信されるファームとキューを表示します。ファームとキューを変更するには、送信者の下部にある設定... ボタンを使用します。
- キュー環境 – キュー環境で定義されたパラメータ値を設定します。Deadline Cloud は DCC アプリケーションのデフォルトのパラメータ値を追加します。必要に応じて値を追加できます。

ジョブ固有の設定タブ

The screenshot shows the 'Submit to AWS Deadline Cloud' dialog box with the 'Job-specific settings' tab selected. The settings are as follows:

Setting	Value
Project Path	C:\Users\user\testCube.blend
Output Directory	C:\Users\user
Output File Prefix	output_####
Scene	Scene
Render Engine	cycles
View Layers	ViewLayer
Cameras	Camera
<input type="checkbox"/> Cycles GPU Rendering	CUDA
<input type="checkbox"/> Override Frame Range	1-250

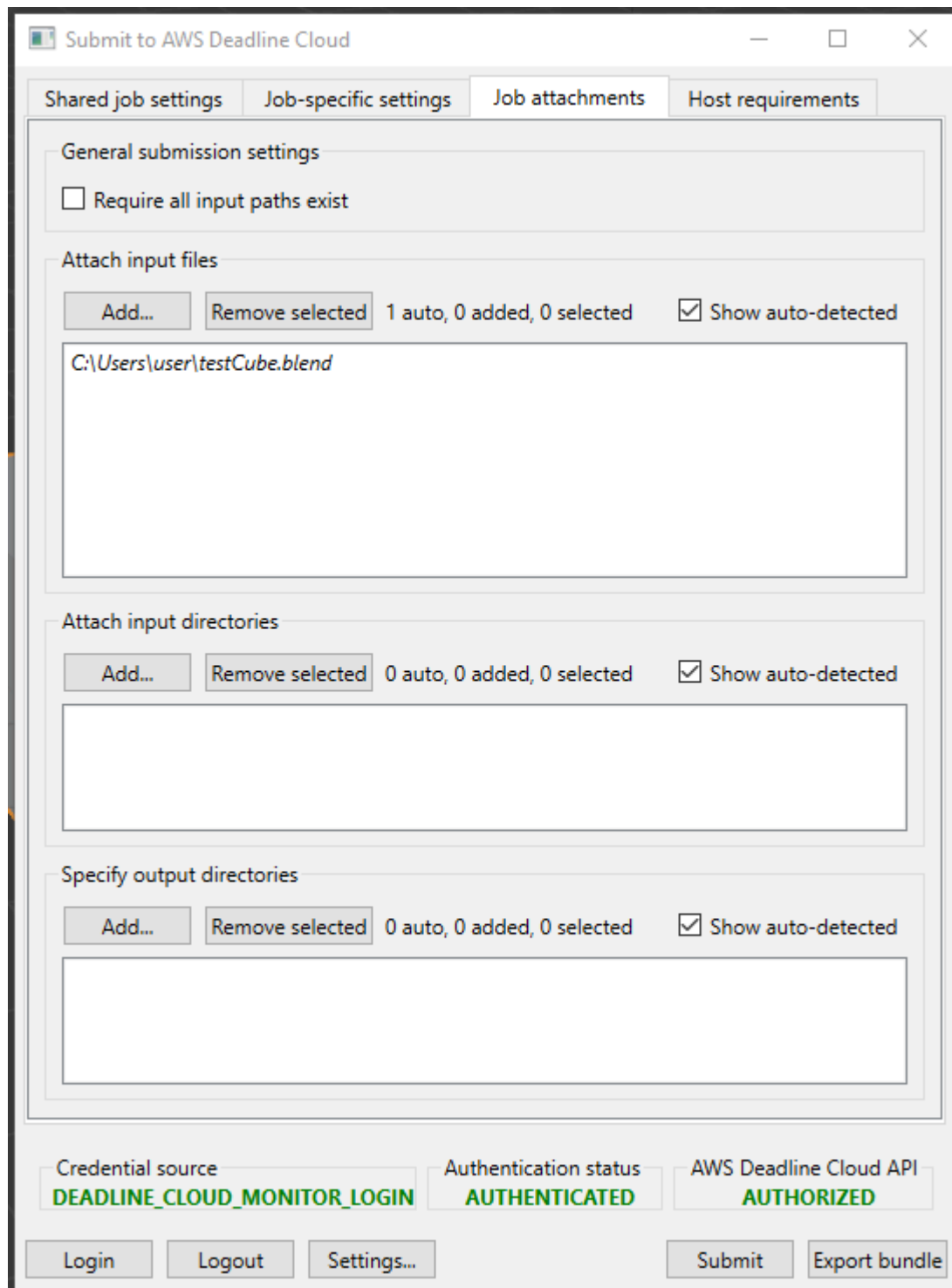
At the bottom of the dialog, the authentication status is shown as:

Item	Status
Credential source	DEADLINE_CLOUD_MONITOR_LOGIN
Authentication status	AUTHENTICATED
AWS Deadline Cloud API	AUTHORIZED

Buttons at the bottom include: Login, Logout, Settings..., Submit, and Export bundle.

ジョブ固有の設定タブには、DCC アプリケーションに固有の設定が含まれています。アプリケーションで使用可能なオプションに基づいて、これらの設定を指定します。

ジョブアタッチメントタブ



ジョブアタッチメントタブには、レンダリングの完了に必要なすべてのファイルが表示されます。送信者は、レンダリングに必要なすべてのファイルを検索しようとします。識別されたファイルは、斜体のリストに表示されます。

自動的に検出されなかったレンダリングに必要な他のアセットを含む入力ファイルとディレクトリを追加できます。

ジョブが複数の出力ディレクトリにファイルを書き込む場合は、ここでディレクトリを指定して、ジョブのダウンロードの一部になるようにする必要があります。

ホスト要件タブ

Submit to AWS Deadline Cloud

Shared job settings Job-specific settings Job attachments **Host requirements**

Run on all available worker hosts

Run on worker hosts that meet the following requirements

All fields below are optional

Operating system -

CPU architecture -

Hardware requirements

vCPUs	Min	-	Max	-
Memory (GiB)	Min	-	Max	-
GPUs	Min	-	Max	-
GPU memory (GiB)	Min	-	Max	-
Scratch space	Min	-	Max	-

Custom host requirements

[More info](#)

Add amount Add attribute

Credential source: DEADLINE_CLOUD_MONITOR_LOGIN Authentication status: AUTHENTICATED AWS Deadline Cloud API: AUTHORIZED

Login Logout Settings... Submit Export bundle

ホスト要件タブは、ジョブの処理に必要なフリート機能を設定します。機能は、フリート内の個々のワーカーではなく、フリート全体に対して指定されます。

キューにリソース制限が関連付けられている場合は、Add amount ボタンを使用して制限を指定します。詳細については、[「ジョブのリソース制限を作成する」](#)を参照してください。

Deadline Cloud ジョブの処理

ジョブがキューに入ると、Deadline Cloud はキューに関連付けられた 1 つ以上のフリートにジョブをスケジューリングします。フリートは、フリート用に設定された機能と特定のステップのホスト要件に基づいて選択されます。ジョブに、キューに関連付けられたフリートのいずれでも満たすことができない要件がある場合、ジョブのステータスは「非互換」に設定され、ジョブの残りのステップはキャンセルされます。

次に、Deadline Cloud はワーカーにステップのセッションを設定する手順を送信します。ステップに必要なソフトウェアは、ジョブを実行するワーカーインスタンスで利用できる必要があります。フリートスケール設定で許可されている場合、サービスは複数のワーカーでセッションを開きます。

Amazon Machine Image (AMI) でソフトウェアをセットアップすることも、ワーカーが実行時にリポジトリまたはパッケージマネージャーからソフトウェアをロードすることもできます。キュー、ジョブ、またはステップ環境を使用して、必要なソフトウェアをデプロイできます。

Deadline Cloud サービスは OpenJD テンプレートを使用して、ジョブに必要なステップと、各ステップに必要なタスクを特定します。一部のステップは他のステップに依存するため、Deadline Cloud はステップを完了する順序を決定します。次に、Deadline Cloud は各ステップのタスクをワーカーに送信して処理します。タスクが完了すると、サービスは同じセッションで別のタスクを送信するか、ワーカーは新しいセッションを開始できます。

各ステップのすべてのタスクが完了すると、ジョブが完了し、出力をワークステーションにダウンロードする準備が整います。ジョブが完了していない場合でも、完了した各ステップとタスクの出力はダウンロードできます。

Note

Deadline Cloud は、ジョブが送信されてから 120 日後にジョブを削除します。ジョブを削除すると、ジョブに関連付けられたすべてのステップとタスクも削除されます。ジョブを再実行する必要がある場合は、ジョブの OpenJD テンプレートを再度送信します。

Deadline Cloud ジョブのモニタリング

AWS Deadline Cloud モニターには、ジョブの全体像が表示されます。これを使用して、次の操作を行います。

- ジョブのモニタリングと管理
- フリートのワーカーアクティビティを表示する
- 予算と使用状況を追跡する
- ジョブの結果をダウンロードします。

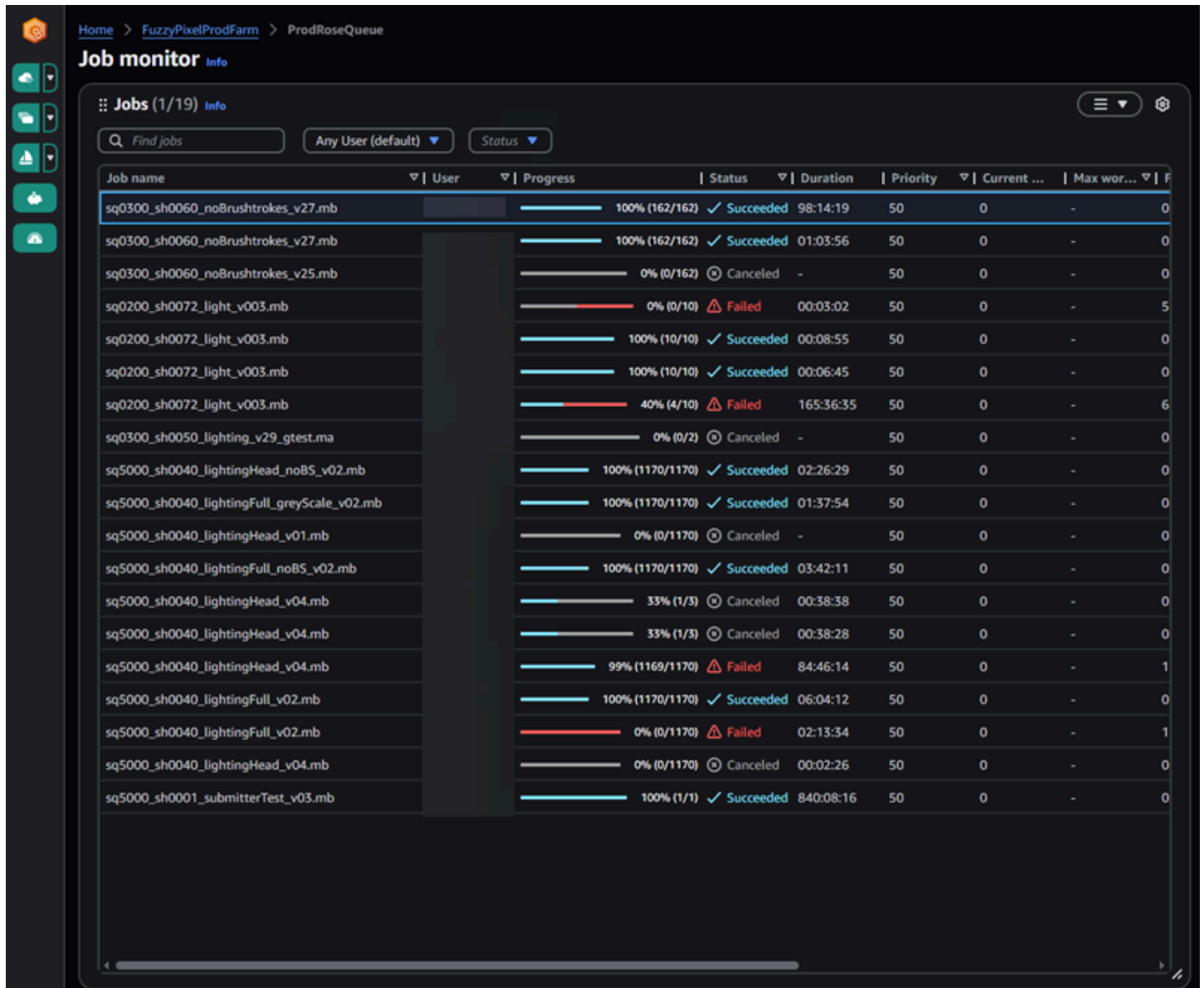
特定のジョブをモニタリングするには、ジョブを含むファームとキューを選択し、リストからジョブを選択します。検索ボックスを使用して、キュー内の特定のジョブを見つけることができます。

ジョブ、ステップ、またはタスクを右クリックすると、項目のオプションが表示されます。以下の操作を実行できます。

- ステータスを変更する
- 項目を停止して再開する
- 項目をキューに入れる
- 出力をダウンロードする
- ジョブの場合: 名前、説明、優先度、最大ワーカー数などのジョブプロパティを変更します。
- タスクの場合: タスクログとワーカーログを表示します。

詳細については、「[Deadline Cloud モニターの使用](#)」を参照してください。

ジョブまたはステップの各タスクにはステータスがあります。ジョブまたはステップのステータスは、タスクのステータスによって異なります。ステータスは、これらのステータスを持つタスクによって順番に決定されます。ステップステータスは、ジョブステータスと同じように決定されます。



The screenshot shows the AWS Deadline Job monitor interface. The top navigation bar includes 'Home > FuzzyPixelProdFarm > ProdRoseQueue'. The main heading is 'Job monitor Info'. Below this, there are filters for 'Jobs (1/19) Info', a search box 'Find jobs', and dropdown menus for 'Any User (default)' and 'Status'. The main content is a table with columns: Job name, User, Progress, Status, Duration, Priority, Current..., and Max wor... (likely Max workers). The table lists 19 jobs with various statuses: Succeeded, Canceled, and Failed. Progress bars and icons (checkmarks, X's, triangles) are used to indicate the status of each job.

Job name	User	Progress	Status	Duration	Priority	Current ...	Max wor...
sq0300_sh0060_noBrushstrokes_v27.mb		100% (162/162)	✓ Succeeded	98:14:19	50	0	-
sq0300_sh0060_noBrushstrokes_v27.mb		100% (162/162)	✓ Succeeded	01:03:56	50	0	-
sq0300_sh0060_noBrushstrokes_v25.mb		0% (0/162)	⊗ Canceled	-	50	0	-
sq0200_sh0072_light_v003.mb		0% (0/10)	⚠ Failed	00:03:02	50	0	5
sq0200_sh0072_light_v003.mb		100% (10/10)	✓ Succeeded	00:08:55	50	0	-
sq0200_sh0072_light_v003.mb		100% (10/10)	✓ Succeeded	00:06:45	50	0	-
sq0200_sh0072_light_v003.mb		40% (4/10)	⚠ Failed	165:36:35	50	0	6
sq0300_sh0050_lighting_v29_gtest.ma		0% (0/2)	⊗ Canceled	-	50	0	-
sq5000_sh0040_lightingHead_noBS_v02.mb		100% (1170/1170)	✓ Succeeded	02:26:29	50	0	-
sq5000_sh0040_lightingFull_greyScale_v02.mb		100% (1170/1170)	✓ Succeeded	01:37:54	50	0	-
sq5000_sh0040_lightingHead_v01.mb		0% (0/1170)	⊗ Canceled	-	50	0	-
sq5000_sh0040_lightingFull_noBS_v02.mb		100% (1170/1170)	✓ Succeeded	03:42:11	50	0	-
sq5000_sh0040_lightingHead_v04.mb		33% (1/3)	⊗ Canceled	00:38:38	50	0	-
sq5000_sh0040_lightingHead_v04.mb		33% (1/3)	⊗ Canceled	00:38:28	50	0	-
sq5000_sh0040_lightingHead_v04.mb		99% (1169/1170)	⚠ Failed	84:46:14	50	0	1
sq5000_sh0040_lightingFull_v02.mb		100% (1170/1170)	✓ Succeeded	06:04:12	50	0	-
sq5000_sh0040_lightingFull_v02.mb		0% (0/1170)	⚠ Failed	02:13:34	50	0	1
sq5000_sh0040_lightingHead_v04.mb		0% (0/1170)	⊗ Canceled	00:02:26	50	0	-
sq5000_sh0001_submitterTest_v03.mb		100% (1/1)	✓ Succeeded	840:08:16	50	0	-

次のリストでは、ステータスについて説明します。

NOT_COMPATIBLE

ジョブ内のタスクの1つを完了できるフリートがないため、ジョブはファームと互換性がありません。

RUNNING

1人以上のワーカーがジョブからタスクを実行しています。実行中のタスクが少なくとも1つある限り、ジョブは `NOT_RUNNING` とマークされます。

ASSIGNED

1人以上のワーカーに、次のアクションとしてジョブ内のタスクが割り当てられます。環境がある場合は、`Environment` がセットアップされます。

STARTING

1人以上のワーカーがタスクを実行するための環境をセットアップしています。

SCHEDULED

ジョブのタスクは、ワーカーの次のアクションとして1つ以上のワーカーにスケジュールされます。

READY

ジョブの少なくとも1つのタスクを処理する準備ができています。

INTERRUPTING

ジョブ内の少なくとも1つのタスクが中断されています。ジョブのステータスを手動で更新すると、中断が発生する可能性があります。また、Amazon Elastic Compute Cloud (Amazon EC2) スポット料金の変更による中断に応じて発生することもあります。

FAILED

ジョブ内の1つ以上のタスクが正常に完了しませんでした。

CANCELED

ジョブ内の1つ以上のタスクがキャンセルされました。

SUSPENDED

ジョブの少なくとも1つのタスクが中断されました。

PENDING

ジョブ内のタスクは、別のリソースの可用性を待っています。

SUCCEEDED

ジョブ内のすべてのタスクが正常に処理されました。

対応ソフトウェア

Deadline Cloud は、3D レンダリング、アニメーション、視覚効果、合成のための幅広いデジタルコンテンツ作成アプリケーションをサポートしています。サポートされているアプリケーションには、常に統合された送信者が含まれますが、conda パッケージ、ホスト設定スクリプト、使用状況ベースのライセンスなどもサポートされる場合があります。以下に示すアプリケーションは、Deadline Cloud からの公式サポートを受けています。公式にサポートされている設定以外のカスタマイズオプションについては、「Deadline Cloud Developer Guide」の「[Provide applications for your jobs](#)」および「[Create a conda package for an application or plugin](#)」を参照してください。

Deadline Cloud では、次の DCC アプリケーションがサポートされています。

トピック

- [Adobe After Effects](#)
- [Autodesk 3ds Max](#)
- [Autodesk Maya](#)
- [Autodesk VRED](#)
- [ブレンダー](#)
- [エピック Unreal エンジン](#)
- [Foundry Nuke](#)
- [KeyShot Studio](#)
- [Maxon シネマ 4D](#)
- [SideFX Houdini](#)

Adobe After Effects

Adobe After Effects は、プロフェッショナルなデジタルビジュアルエフェクト、モーショングラフィックス、合成アプリケーションです。After Effects は Deadline Cloud で完全にサポートされており、送信者や conda パッケージなどの包括的な統合によりレンダリングパフォーマンスが向上します。このガイドでは、Deadline Cloud with After Effects AWS を使用して、レンダリングタスクを複数のマシンに分散することでプロジェクトをより迅速にレンダリングする step-by-step について説明します。

サポートの概要

After Effects は、次のコンポーネントでサポートされています。

- 送信者: シーンとアセットの自動検出を使用して After Effects から直接ジョブを送信するための統合送信者。
- Conda パッケージ: サービスマネージドフリートへの自動インストールの Deadline Cloud。
- クロスプラットフォーム互換性: Windows および macOS の送信者サポートと Windows のワーカーサポート。

After Effects バージョンの互換性

次の表は、After Effects バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート
2024	Windows、macOS	Windows
2025	Windows、macOS	Windows
2026	Windows、macOS	Windows

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルでサービスマネージドフリートが利用できる After Effects に適用されるすべての conda パッケージを示します。

OS	パッケージ	バージョン
Windows	後続効果	24.6
Windows	後続効果	25.1
Windows	後続効果	25.2
Windows	後続効果	25.6
Windows	後続効果	26.0

開始方法

Deadline Cloud で After Effects を設定するには、次の手順を実行します。必要な送信者とモニターをワークステーションにインストールし、レンダリングジョブをキューに送信し始めます。

1. サービスマネージドフリートを作成し、キューに関連付けます。キューは、期限クラウド conda チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。
2. Deadline Cloud モニターインストーラを使用して、アーティストワークステーションに Deadline Cloud モニターをインストールします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. Deadline Cloud Submitter Installers を使用して、アーティストワークステーションに Deadline Cloud After Effects 送信者をインストールします。送信者をインストールするときは、ユーザーインストール (管理者不要) またはシステムインストール (Windows のみ、管理者が必要) のいずれかを選択できます。macOS ユーザーはユーザーインストールを使用する必要があります。
 - ユーザーインストール: 管理者権限なしでユーザーディレクトリにインストールします。送信者は、ドッキング可能なパネルではなくスタンドアロンウィンドウになります。
 - Windows: `C:\Users\\DeadlineCloudSubmitter\Submitters\AfterEffects\AE<version>`
 - macOS: `/Users/<user>/Library/Preferences/Adobe/After Effects/<version>/Scripts/ScriptUI Panels`
 - System Install (Windows のみ): ドッキング可能なパネルとして Adobe After Effects インストールディレクトリにインストールします。
 - Windows: `C:\Program Files\Adobe\Adobe After Effects <version>\Support Files\Scripts\Script UI Panels`

インストール

After Effects の Deadline Cloud 送信者をインストールするには、次の環境を準備します。

- Windows または macOS ワークステーション。
- Adobe After Effects 24、25、または 26 のインストール。
- Python 3.9 以降。
- 次のいずれかを使用して Deadline Cloud ファームにアクセスします。
 - After Effects が利用可能なサービスマネージドフリート (カスタム conda パッケージ経由)。

- After Effects とライセンスが設定されているカスターマネージドフリート。

前提条件

送信者をインストールする前に、次の手順を実行します。

1. Deadline Cloud の送信者を実行して Deadline Cloud CLI と Deadline Cloud モニターをインストールし、AWS コンソールの Deadline Cloud サービスのダウンロードセクションからインストーラをモニタリングします。
2. After Effects でスクリプトのアクセス許可を有効にします。送信者は、適切に機能するために、ファイルを書き込み、ネットワーク経由で通信を送信できる必要があります。デフォルトでは、After Effects スクリプトはこれらのアクションを実行できません。詳細については、[「Adobe のスクリプトリファレンス」](#)を参照してください。

スクリプトがファイルを書き込んだり、ネットワーク経由で通信を送信したりできるようにするには、オペレーティングシステムの手順を実行します。

- Windows: Edit > Preferences > Scripting & Expressions > Select Scripts to Write Files and Access Network を選択します。
- macOS: After Effects > Settings > Scripting & Expressions > Select Scripts to Write Files and Access Network を選択します。

送信者のインストール

After Effects 送信者拡張機能を使用すると、After Effects 内から直接 Deadline Cloud にジョブを送信できます。

Note

インストーラプロセスでは、ユーザーインストールまたはシステムインストールのいずれかを選択します。macOS ユーザーはデフォルトでユーザーインストールになります。これは、自動システムインストールは現在 macOS ではサポートされていないためです (手動システムインストールは管理者ユーザーでも可能です)。選択内容に一致する以下のセクションのみをお読みください。

ユーザーインストール

1. [ステップ 1: Deadline Cloud Submitter をインストールして、Deadline Cloud Submitter インストーラをダウンロードします。](#)
2. インストーラを実行します (管理者は必要ありません)。
3. プロンプトに従って After Effects 送信者を選択し、メジャーバージョン (24、25、26 など) を選択します。
4. 送信者はオペレーティングシステムに基づいてインストールされます。
 - macOS: インストーラは、選択したメジャーバージョンのすべてのマイナーバージョンの After Effects ユーザー設定ディレクトリ `DeadlineCloudSubmitter_Assets` に `DeadlineCloudSubmitter(User).jsx` とを自動的に配置します。

```
/Users/<user>/Library/Preferences/Adobe/After Effects/<version>/Scripts/ScriptUI  
Panels
```

送信者は、After Effects のウィンドウメニューに表示されます。

- Windows: デフォルトでは、送信者はスタンドアロンの場所にインストールされます。

```
C:\Users\<user>\DeadlineCloudSubmitter\Submitters\AfterEffects\AE<version>
```

ファイル > スクリプト > スクリプトファイルの実行を使用して、スクリプトを手動で実行する必要があります。

システムのインストール

Note

自動 macOS システムインストールは、送信者インストーラではまだサポートされていません。macOS 管理者ユーザーは、アプリケーションディレクトリにファイルをコピーして手動システムインストールを実行できます (「」を参照[手動インストール \(代替\)](#))。

1. [ステップ 1: Deadline Cloud Submitter をインストールして、Deadline Cloud Submitter インストーラをダウンロードします。](#)
2. Windows のみ: インストーラを右クリックし、管理者として実行を選択します。
3. プロンプトに従って、After Effects 送信者を選択します。インストール先:

- Windows: C:\Program Files\Adobe\Adobe After Effects <version>\Support Files\Scripts\Script UI Panels
- macOS (手動): /Applications/Adobe After Effects <version>/Scripts/ScriptUI Panels

最終セットアップ手順

1. 送信者インストーラを使用してインストールしたら、ターミナル/PowerShell `deadline auth login` を実行するか、Deadline Cloud モニターを使用してログインして、Deadline Cloud ユーザープロファイルにログインします。
2. After Effects 送信者が使用する必要な依存関係をインストールするには、ローカルターミナルまたはコマンドプロンプトで以下を実行します。

```
pip install fonttools
```

3. After Effects が開いている場合は再起動します。

手動インストール (代替)

を手動でインストールする場合は、送信者ファイルを直接コピーできます。

1. [deadline-cloud-for-after-effects](#) リポジトリの `DeadlineCloudSubmitter_Assets` フォルダにある `DeadlineCloudSubmitter.jsx` ファイルと `dist` フォルダを見つけます。
2. After Effects インストール内の `ScriptUI Panels` フォルダに両方をコピーします。
 - Windows: Program Files\Adobe\Adobe After Effects <version>\Support Files\Scripts\Script UI Panels
 - macOS: Applications/Adobe After Effects <version>/Scripts/Script UI Panels
3. After Effects が開いている場合は再起動します。

Deadline Cloud フォームで After Effects をセットアップする

After Effects conda パッケージが Deadline Cloud のサービスマネージドフリートで利用可能になりました。サポートされているバージョンのリストについては、[Deadline Cloud ユーザーガイド](#)を参照してください。別の After Effects conda パッケージを含む conda チャンネルを構築する場合は、[カスタム conda チャンネルを作成する手順に従ってください](#)。

パッケージを構築する際のリファレンスとして、[deadline-cloud-samples パッケージ](#)の After Effects conda レシピを使用することもできます。

この送信者によって作成されたジョブでは、ジョブを実行するユーザーの PATH aerender で実行可能ファイルを使用する必要があります。または、アレンダー実行可能ファイルを指すように AERENDER_EXECUTABLE 環境変数を設定することもできます。

送信者の更新

送信者を最新バージョンに更新するには、最新の送信者インストーラをダウンロードして実行します。

After Effects 送信者の使用

After Effects に Deadline Cloud 送信者を使用するには、ファームに After Effects 対応フリートが設定されていることを確認し (「」を参照[Deadline Cloud ファームで After Effects をセットアップする](#))、送信者がインストールされていることを確認します。Deadline Cloud モニターにログインするか、Deadline Cloud アクセスの設定プロファイルを使用して AWS 認証情報を指定します。

ジョブを送信する

Note

インストール時に選択したインストールタイプに一致する以下の手順に従ってください。

ユーザーインストール - macOS

1. Adobe After Effects を起動します。
2. Window > DeadlineCloudSubmitter(User).jsx を選択して送信者を開きます。
3. [ジョブの送信](#) 以下のステップに従います。

ユーザーインストール - Windows

1. Adobe After Effects を起動します。
2. ファイル > スクリプト > スクリプトファイルの実行を選択して、送信者を開きます。DeadlineCloudSubmitter.jsx ファイルに移動し、ファイルを選択して送信者を実行します。最近送信者を閉じた場合は、ファイル > スクリプト > 最近のスクリプトファイルを選択し、DeadlineCloudSubmitter.jsx ファイルを選択して、送信者を再度開くことができます。

3. [ジョブの送信](#) 以下のステップに従います。

システムのインストール

1. After Effects を管理者 (Windows) または適切なアクセス許可 (macOS) で起動します。
2. Window > DeadlineCloudSubmitter.jsx を選択して送信者を開きます。
3. [ジョブの送信](#) 以下のステップに従います。

ジョブの送信

1. 送信者の Open Render Queue を選択します。任意のコンポジションをレンダーキューに追加し、レンダー設定、出力モジュール、出力パスを設定します。
2. 送信者で更新を選択して、コンポジションリスト内のコンポジションを表示します。
3. レンダリングするコンポジションを選択し、送信を選択してレンダリングジョブを送信します。使用可能な設定の詳細については、[After Effects 固有の設定](#)以下を参照してください。
4. 「ファイルに含まれるスクリプトを実行しようとしています」という警告ポップアップウィンドウが表示された場合は、ポップアップの指示に従って警告を抑制することができます。
5. プロンプトが表示されたら Python ライブラリをインストールし、ログインしていない場合は左下のログインボタンを選択します。
6. 設定ボタンを使用して送信先のファームとキューを設定し、送信を選択します。

Note

After Effects 送信者は Deadline Cloud GUI 送信者を呼び出してジョブ送信を完了します。GUI 送信者で問題が発生した場合は、[「 deadline-cloud library 」](#)を参照してください。

共有ジョブ設定

以下の設定がジョブ全体に適用されます。

- ファームの選択 - ジョブがレンダリングするファームを選択します。
- キューの選択 - 選択したファーム内の特定のキューを選択します。
- ジョブ名 - レンダリングジョブにわかりやすい名前を付けます。
- ジョブの説明 - レンダリングジョブに関するオプションの詳細を追加します。

- Priority - キュー管理のジョブ優先度を設定します。
- 初期状態 - ジョブをすぐに開始するか、一時停止したままにするかを制御します。
- 最大失敗タスク数 - ジョブが失敗とマークされるまでに失敗する可能性があるタスクの最大数を設定します。
- タスクあたりの最大再試行回数 - 失敗したタスクが再試行される回数を設定します。
- 最大ワーカー数 - このジョブで同時に作業できるワーカーの最大数を設定します。

After Effects 固有の設定

以下の設定は、After Effects レンダリングに固有です。

- コンポジションの選択 - 送信するレンダーキューからコンポジションを選択します。
- タスクあたりのフレーム数 - (イメージシーケンスの場合) 各タスクがレンダリングするフレーム数を指定します。デフォルトでは、Deadline Cloud はフレームごとに 1 つのタスクを作成します。この値を大きくすると、各フレームがすばやくレンダリングされるジョブのレンダリングが高速化されます。
- マルチフレームレンダリング - After Effects マルチフレームレンダリングを有効にします。レンダリングに割り当てる CPU 使用率の最大パーセンテージを指定することもできます。詳細については、[Adobe のマルチフレームレンダリングドキュメント](#)を参照してください。
- タイムアウト - (オプション) タスクが応答しなくなる可能性のある予期しないケースを処理するようにタイムアウト設定 (日、時間、分) を設定します。デフォルトのタイムアウトは 2 日です。
- 出力タイプ - レンダリングキューの設定 (イメージシーケンスまたはビデオ) に基づいて自動的に検出されます。
- 出力パス - レンダリングされたファイルが保存されるディレクトリ (レンダリングキュー設定から)。

他の送信者タブの詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

フォントアタッチメントシステム

送信されたコンポジションで使用されるフォントは、送信者によって検出され、送信時にジョブの添付ファイルとして自動的に追加されます。フォントは、レンダリングの開始前にワーカーにインストールされ、ジョブが終了すると削除されます。

サポートされているフォントタイプ:

- OpenType (.otf)。
- TrueType (.ttf)。
- TrueType コレクション (.ttc) - 過半数をサポート。
- [Adobe フォント](#)。
- Windows ビットマップフォント (.fon) - Windows マシンでのみサポートされています。

フォントのトラブルシューティング:

レンダリング時にフォントがない場合:

1. システムにフォントがインストールされていることを確認します。
 - Windows: 設定を開く > パーソナライズ > フォント インストールされたフォントを表示します。
 - macOS: Font Book アプリケーションを開いて、インストールされたフォントを表示します。
2. フォントがジョブアタッチメントに含まれていることを確認する: 送信者で送信を選択したら、送信ダイアログのジョブアタッチメントタブをチェックして、フォントがリストされていることを確認します。

Adobe Creative Cloud フォント:

Deadline Cloud が Adobe Creative Cloud を介して配布されるフォントを使用するには、ワークステーション上のすべての Adobe 以外のアプリケーションでフォントを使用できるようにする必要があります。

Creative Cloud に Adobe 以外のアプリケーションのフォントをインストールするには:

1. Adobe Creative Cloud Desktop を開きます。
2. 計画の下のアカウントサイドバーで Adobe Fonts を選択し、Adobe Fonts パネルを表示します。
3. Adobe Fonts サイドバーで追加されたフォントを選択して、追加されたフォントを表示します。
4. Adobe 以外のアプリケーションで使用できるようにするフォントの横にあるインストールファミリーを選択します。

ジョブバンドルの表示

ジョブを送信するには、送信者がまず [ジョブバンドル](#) を生成し、次に deadline-cloud パッケージの機能を使用してジョブバンドルをレンダーファームに送信して実行します。

ファームに送信されるジョブを表示する場合:

1. 送信者のバンドルのエクスポートボタンを使用して、ジョブ履歴ディレクトリのジョブバンドルをエクスポートします (デフォルト: `~/.deadline/job_history`)。
2. 送信者ではなくエクスポートからジョブを送信する場合は、Deadline Cloud アプリケーションを使用して、そのバンドルをファームに送信できます。

トラブルシューティング

以下のセクションでは、Deadline Cloud での After Effects 送信者の使用に関する一般的な問題と質問について説明します。

質問のレンダリング

Q: Deadline Cloud ワーカーでカスタム ExtendScript (.jsx) スクリプトを実行できますか?

A: はい、制限があります。サービスマネージドフリートの After Effects conda パッケージは、After Effects の完全なインストールではなく、レンダラーとその依存関係をパッケージ化します。ホスト設定を使用して、ワーカーでカスタム .jsx スクリプトを実行できます。ただし、ネットワーク呼び出しを行うスクリプトはワーカーでは機能しません。これは、「ファイルの作成とネットワークへのアクセスをスクリプトに許可する」設定は、フルインストールなしでは設定できないローカルユーザー設定であるためです。

トラブルシューティングに関する質問

Q: After Effects ジョブがタイムアウトしているか、セッションのクリーンアップ中に PermissionDenied エラーが表示されます。何が起きているのでしょうか?

A: この問題は通常、ヘッドレスモードで却下できない UI ダイアログ (、 など `confirm()`) が原因で After Effects `alert()` がハングアップしたことが原因で発生します。症状は次のとおりです。

- タイムアウトエラー: プロセスは、ユーザーによる操作を待つ間ハングし、`subprocess.TimeoutExpired` エラーが発生します。
- PermissionDenied errors during session cleanup: The hung After Effects process holds locks on Adobe DLLs (AdobeXMP.dll、dvacore.dll、dynamiclink.dll など)。ワーカーがセッションのクリーンアップを試みると、Access to the path '<file>' is denied エラーが発生します。

根本原因 (ハングプロセス) を修正すると、タイムアウトエラーとクリーンアップエラーの両方が解決されます。ファーム実行用のスクリプトからすべての UI 呼び出しを削除します。

トラブルシューティングのヒント: After Effects UI を使用して Windows ワークステーションで同じワークフローを実行して、Deadline クラウド固有の問題を排除してみてください。

Q: ジョブの実行時に「aerender Error: Could not read from source」が表示されます。何が問題ですか？

A: このエラーは通常、ジョブを送信する前にコンポジションのレンダーキューに出力パスが設定されていないことを意味します。After Effects 送信者は、コンポジションをレンダーキューに追加し、送信前にレンダー設定、出力モジュール、出力パスを設定する必要があります。完全な送信ワークフロー [After Effects 送信者の使用](#) については、「」を参照してください。

Q: での「ワーカーへのフォントのインストール」でジョブが失敗します **AddFontResource failed to load**。どうすればよいですか？

A: OS 固有の制限により、一部のフォントは特定のオペレーティングシステムにインストールできません。フォントマネージャーがワーカーにサポートされていないフォントをインストールしようとすると、次のようなエラーで失敗します。

```
OSError: AddFontResource failed to load "...\\tempFonts\\HelveticaNeue-CondensedBlack.ttc"
```

この問題を解決するには、次の手順を試してください。

1. フォントが実際にコンポジションで使用されているかどうかを確認します。誤ってアップロードされた可能性があります。
2. フォントが必要ない場合は、再送信する前に送信者のジョブアタッチメントから削除します。
3. フォントが必要な場合は、Windows と互換性のある別のフォントを置き換えてみてください (Deadline Cloud は Windows ワーカーへの影響後にのみ実行されます)。
4. 回避策として、テキスト [からシェイプを作成を使用して After Effects でテキスト](#) レイヤーをシェイプに変換することで、フォントの依存関係を完全に削除できます。この変換は一方向であることに注意してください。テキストを編集したり、テキスト固有の機能を使用したりすることはできなくなります。
5. 上記のオプションがいずれも機能しない場合は、[リポトリで問題を作成して](#) 詳細な調査を行います。

ヒント: Windows マシンのフォントファイルをダブルクリックすることで、フォントが Windows にインストール可能かどうかを確認できます。開き、「インストール」オプションが表示される場合、フォントがサポートされます。

エラー: PATH で Python 3 以上が見つかりませんでした

macOS の場合:

1. ターミナルを開き、`where python`および `where python3` のコマンドを実行します。`where python3`。結果が得られない場合は、ワークステーションに Python をインストールする必要があります。
2. `python3 CLI` がインストールされていないが、`python`がある場合は`python --version`を実行して Python 3.9 以降があるかどうかを確認します。そうでない場合は、最初に Python 3.9 以降をインストールし、パスに追加します。
3. バージョンチェックの結果とチェック場所を取得したら、どの Python 実行可能ファイルがアクティブに使用されているかを確認します。`which python`と `which python3` を実行します。`which python3`。結果が得られない場合は、Python CLI を `zsh` に追加する必要があります`$PATH`。
4. また、Deadline Cloud CLI のインストールに使用された Python を追加していることを確認する必要があります。`python -m pip list` および `python3 -m pip list` を実行して検証し、`$PATH` 該当する Python に追加します。
5. `~/.zshrc` (および `~/.bashrc` 該当する場合は更新) して、Python CLI を含む `bin` フォルダをパスに追加します`$PATH`。たとえば、Python および Deadline Cloud CLI が `/Library/Frameworks/Python.framework/Versions/3.13/bin` の下にある場合は `~/.zshrc` `$PATH` が評価されたときに最終的な優先度になるように、ファイルの最後に次の行をファイルに追加します。

```
export PATH=$PATH:/Library/Frameworks/Python.framework/Versions/3.13/bin
```

6. 保存して終了し、`source ~/.zshrc` を実行します。
7. `python --version` と `deadline --version` を実行すると両方の実行可能ファイルと After Effects にもアクセスできます。ジョブの送信を再試行します。出力は次のようなパターンに従う必要があります。

```
user@7cf34df03377 ~ % which python3
/Library/Frameworks/Python.framework/Versions/3.13/bin/python3
user@7cf34df03377 ~ % where deadline
/Library/Frameworks/Python.framework/Versions/3.13/bin/deadline
```

Windows の場合:

1. コマンドプロンプト (または PowerShell) を開きます。
2. `where python`、`where python3`、`where py`、および `where` を実行して、実行可能ファイル `where deadline` を見つけます。PowerShell では、`Get-Command python` の代わりに `where python` を使用します。
3. Windows + S を押して、「システム環境変数の編集」を検索します。このステップには管理者アクセスが必要です。環境変数...を選択します。
4. Deadline Cloud CLI がユーザーインストールかシステムインストールかに応じて、ユーザー変数またはシステム変数のいずれかでパス変数を選択し、編集を選択します。
5. 編集ウィンドウで、新規を選択し、Python および Deadline Cloud 実行可能ファイルを含むフォルダへのパスを貼り付けます。
6. 期限と Python CLI を含むバイナリフォルダが追加されていることを確認します。Deadline は、送信者インストーラからインストールする場合は `DeadlineCloudSubmitter` フォルダに、`pip` で管理する場合は Python フォルダにあります。`pip` で Deadline Cloud CLI を管理している場合は、`python -m pip list` および/または `python3 -m pip list` を実行して、Deadline Cloud CLI を含む Python をパスに追加します。

エラー: Deadline Not Found

1. ターミナルまたはコマンドプロンプトを開き、`deadline --version` を実行してインストールを確認します。
2. 次に、オペレーティングシステムの Python に関する上記のトラブルシューティング手順に従い、期限が `$PATH` にあることを確認します。
3. 複数の Python インストールがあり、`pip` を使用して Deadline Cloud CLI を管理する場合は、`$PATH` の Python が Deadline Cloud CLI インストールを管理する Python であることを確認します。`python -m pip list` と `python3 -m pip list` を実行して確認します。

エラー: ジョブテンプレートがありません

エラーの例: `Missing job template at /Applications/Adobe After Effects 2025/Scripts/ScriptUI Panels/DeadlineCloudSubmitter_Assets/JobTemplate.`

このエラーは、`DeadlineCloudSubmitter_Assets` フォルダが見つからないか、`DeadlineCloudSubmitter.jsx` スクリプトファイルの横にない場合に発生します。

1. DeadlineCloudSubmitter.jsx スクリプトのインストール先または移動先を見つけます。
2. DeadlineCloudSubmitter_Assets フォルダがスクリプトと同じディレクトリにあることを確認します。
3. アセットフォルダがない場合は、元のインストールソースから JSX スクリプトと同じ場所にコピーします。
4. After Effects を再起動し、送信者を再度実行してみてください。

警告: サポートされていない After Effects バージョンが検出されました

この警告は、期限クラウド conda チャンネルで利用できない After Effects バージョンを使用していることを意味します。例えば、After Effects 24.3 を使用している場合、チャンネルは 24.6 のみをサポートします。

続行すると、ローカルで使用されているデフォルトのメジャーバージョンが CondaPackages フィールドに入力されますが、特定の After Effects バージョンでカスタム conda チャンネルを作成し、このチャンネルを CondaChannels パラメータに含めない限り、ジョブの送信が失敗する可能性があります。

この問題は、以下のいずれかの方法で解決できます。

1. サポートされている After Effects バージョンに切り替えます。
2. 警告を確認し、続行します (自分の責任で)。
3. 目的の After Effects バージョンを使用してカスタム conda チャンネルを作成します。

送信後に Windows でコマンドプロンプトが点滅する

Windows のスタートメニューに移動し、「アプリケーション実行エイリアスの管理」を検索します。次に、エイリアス python3.exe と python.exe エイリアスを手動で無効にし、送信を再試行します。

サポートされていない拡張子を持つフォントが見つかりました

サポートされているフォントタイプ [フォントアタッチメントシステム](#) については、「」を参照してください。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアのバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとしています。これは、ワークステーションバージョンの After Effects が上記のバージョンテーブルに表示されない場合に失敗します。

サポートされていないバージョンの After Effects が必要な場合は、次のオプションがあります。

- After Effects からジョブを送信する場合、CondaPackages キューパラメータを上書きして、ワーカーで使用するサポートされているバージョンを指定できます (例: `aftereffects=2025`)。これは、シーンで使用される機能と、After Effects がワークステーションバージョンのシーンとどのように連携するかに応じて、機能する場合と機能しない場合があります。
- ワーカーにインストールする目的のバージョンのカスタム conda レシピとチャンネルを構築できます。以下でリンクされているサポートされているバージョンの conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

オープンソースリソース

送信者はオープンソースであり、GitHub で利用できます。

- [効果後の期限クラウド](#)
- [Standalone After Effects ジョブバンドル](#)は GitHub で利用できます。

Autodesk 3ds Max

Note

Deadline Cloud で Autodesk 3ds Max AWS を使用する場合、サブスクリプションに含まれている Autodesk Cloud Rights を使用できます。Cloud Rights とサブスクリプションのメリットの詳細については、Autodesk ウェブサイトの [「Subscription Benefits FAQ: Cloud Rights」](#)を参照してください。

Autodesk 3ds Max は、3D アニメーション、モデル、ゲーム、イメージを作成するためのプロフェッショナル 3D コンピュータグラフィックスプログラムです。Deadline Cloud は、統合された送信者、ホスト設定スクリプト、使用状況ベースのライセンス、およびレンダリングパフォーマンスを向上させるアダプターを備えた 3ds Max の包括的なサポートを提供します。このガイドでは、Deadline Cloud AWS with 3ds Max を使用して、レンダリングタスクを複数のマシンに分散することでプロジェクトをより迅速にレンダリングするためのstep-by-stepの手順を示します。

サポートの概要

3ds Max は、次のコンポーネントでサポートされています。

- 送信者: シーンとアセットの自動検出を使用して 3ds Max から直接ジョブを送信するための統合送信者。
- ホスト設定スクリプト: 3ds Max をインストールするホスト設定スクリプトの例。
- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows の送信者サポートと Windows のワーカーサポート、および自動パスマッピング。
- 使用量ベースのライセンス: Pay-as-you-goのライセンス。

3ds Max バージョンの互換性

次の表は、3ds Max バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	ホスト設定のサポート
2024	Windows	Windows
2025	Windows	Windows
2026	Windows	Windows

3ds 他のデジタルコンテンツ作成ツールとの最大の違い

Deadline Cloud では、3ds Max は conda パッケージの代わりにホスト設定スクリプトを使用してインストールされます。これは、システム管理者がアプリケーションをインストールする必要があります。

るため、3ds Max のインストールプロセスの固有の要件により、Deadline Cloud の他のほとんどの DCCs とは異なります。

開始方法

Deadline Cloud で 3ds Max を使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。GPU アクセラレーションレンダリング機能を使用する場合は、GPU サポートを使用してフリートを設定します。フリートは、3ds Max をインストールするホスト設定スクリプトで設定する必要があります。詳細については、GitHub の「[3ds Max Host Configuration script setup](#)」と「3ds Max Host Config example」を参照してください。 [GitHub](#)
2. Deadline Cloud Submitter を使用してアーティストワークステーションに Deadline Cloud モニターと 3ds Max 送信者をインストールし、インストーラをモニタリングします。詳細については、「[ワークステーションのセットアップ](#)」を参照してください。
3. 統合された送信者を使用して 3ds Max から キューにジョブを直接送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

フリートホスト設定

3ds Max 送信者を設定する前に、Deadline Cloud フリートを次のように設定します。

3ds Max は、Autodesk が提供する一般的なデジタルコンテンツ作成ツールです。3ds Max は Windows で実行され、ホストにインストールするには管理アクセスが必要です。管理要件のため、Deadline Cloud では、ホスト設定スクリプトを使用してワーカーホストに 3ds Max をインストールすることをお勧めします。

[カスタムフリートホスト設定スクリプト](#)を使用すると、サービスマネージドフリートワーカーでソフトウェアのインストールなどの管理タスクを実行できます。これらのスクリプトは昇格された権限で実行されるため、システムのワーカーを柔軟に設定できます。

例

例は、3ds Max 2024 および 2025 と V-Ray、および tyFlow などのプラグインとの統合で使用できます。追加の例をリクエストするには、[ディスカッションフォーラム](#)でアイデアを提案してください。

完全なホスト設定スクリプトの例については、「[deadline-cloud-samples/host_configuration_scripts/3dsmax](#)」を参照してください。

Note

この例では特定の 3ds Max バージョンをインストールしていますが、Deadline Cloud 送信者は 3ds Max 2025 および 2026 もサポートしています。インストールスクリプトは、3ds Max 2025 および 2026 で同等に動作します。

インストール

Autodesk 3ds Max の Deadline Cloud 送信者をインストールするには、次の環境を準備します。

- Windows ワークステーション。
- Autodesk 3ds Max 2024、2025、または 2026 のインストール。
- オプション: 3ds Max インストール用の V-Ray 6 または 7。
- 次のいずれかを使用して Deadline Cloud ファームにアクセスします。
 - Autodesk 3ds Max ホスト設定の Windows サービスマネージドフリート。
 - Autodesk 3ds Max とライセンス設定によるカスタマー管理フリート。

送信者のインストール

Autodesk 3ds Max 送信者拡張機能を使用すると、3ds Max 内から直接 Deadline Cloud にジョブを送信できます。送信者をインストールするには:

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラを実行し、画面の指示に従います。
3. インストール後に 3ds Max を起動します。

送信者の更新

送信者を最新バージョンに更新するには、最新の送信者インストーラをダウンロードして実行します。

Autodesk 3ds Max 送信者の使用

3ds Max の Deadline Cloud 送信者を使用するには、ファームに 3ds Max 対応フリートが設定されていること、および送信者がインストールされていることを確認します。Deadline Cloud モニターに

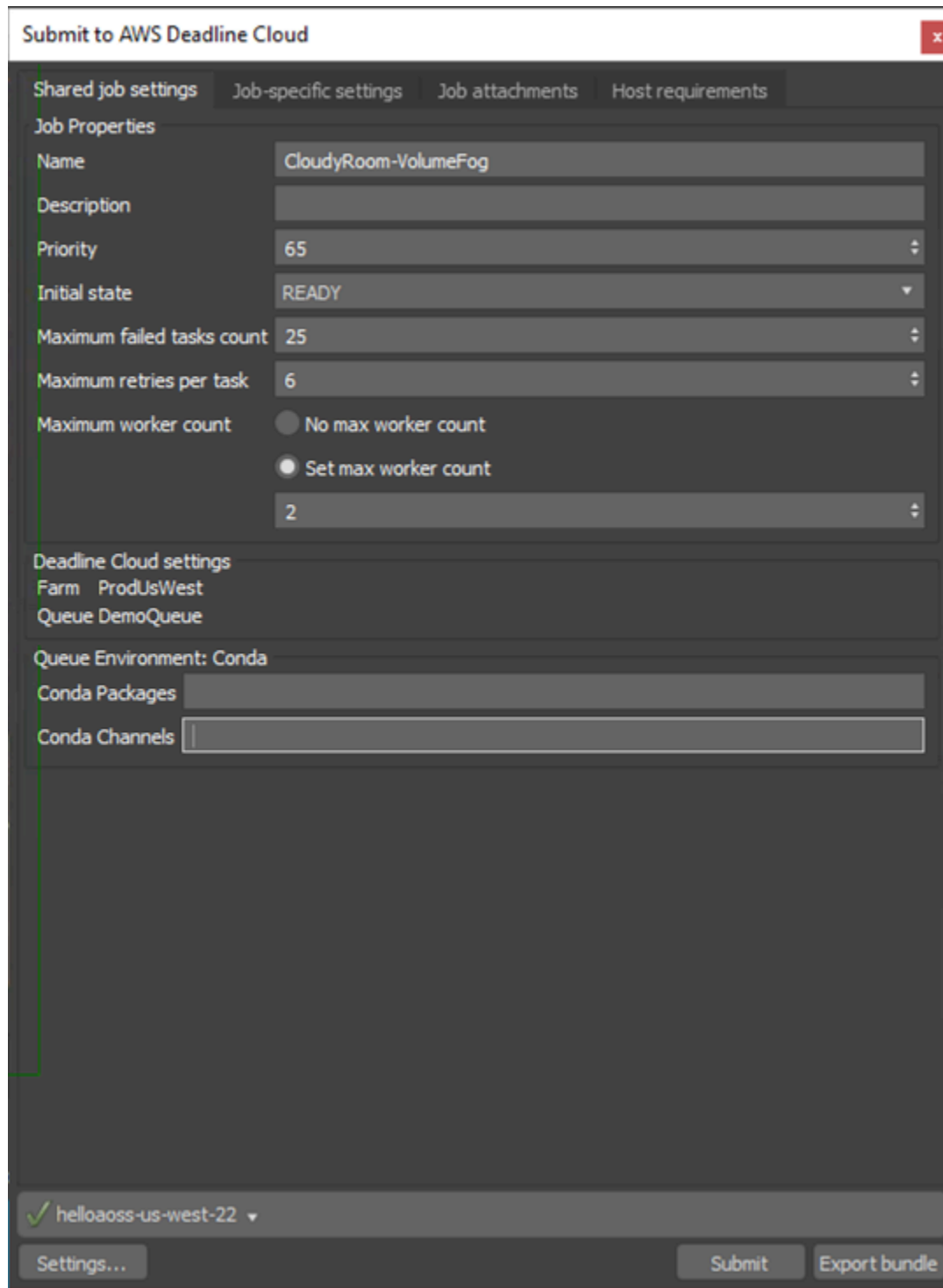
ログインするか、Deadline Cloud アクセスの設定プロファイルを使用して AWS 認証情報を指定します。

ジョブの送信

3ds Max から Deadline Cloud にジョブを送信するには:

1. 3ds Max ファイルを保存します。
2. 3ds Max メニューバーで、Deadline Cloud を選択します。
3. ダイアログのタブを使用してジョブをカスタマイズします。
4. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
5. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

共有ジョブ設定



The screenshot shows a dialog box titled "Submit to AWS Deadline Cloud" with a close button (X) in the top right corner. The dialog is divided into four tabs: "Shared job settings" (selected), "Job-specific settings", "Job attachments", and "Host requirements".

Job Properties

Name	CloudyRoom-VolumeFog
Description	
Priority	65
Initial state	READY
Maximum failed tasks count	25
Maximum retries per task	6
Maximum worker count	<input type="radio"/> No max worker count <input checked="" type="radio"/> Set max worker count 2

Deadline Cloud settings

Farm ProdUsWest
Queue DemoQueue

Queue Environment: Conda

Conda Packages
Conda Channels

helloaoss-us-west-22

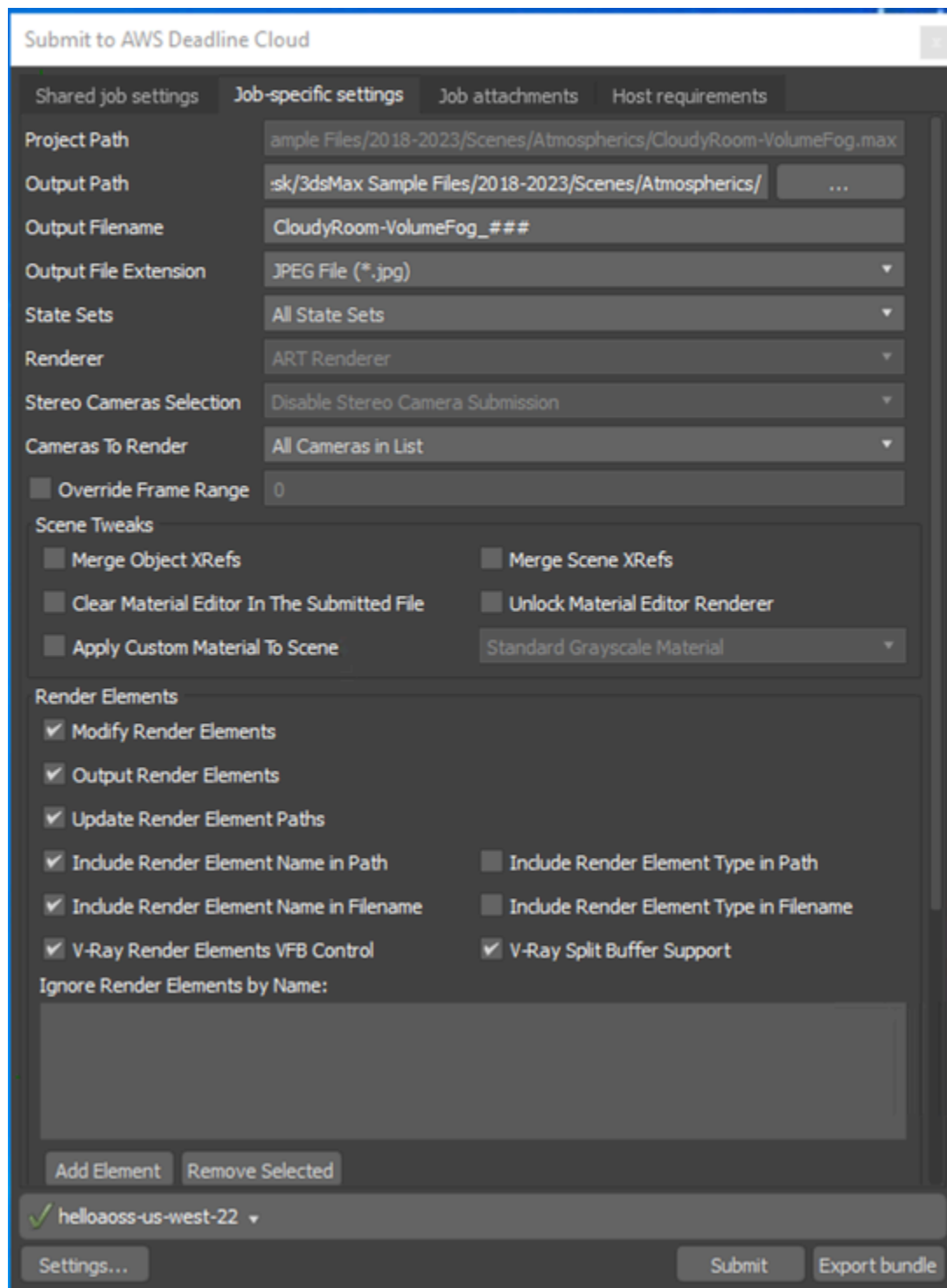
Settings... Submit Export bundle

以下の設定がジョブ全体に適用されます。

- ファームの選択 - ジョブがレンダリングするファームを選択します。
- キューの選択 - 選択したファーム内の特定のキューを選択します。

- ジョブ名 - レンダリングジョブにわかりやすい名前を付けます。
- ジョブの説明 - レンダリングジョブに関するオプションの詳細を追加します。
- Priority - キュー管理のジョブ優先度を設定します。
- 初期状態 - ジョブをすぐに開始するか、一時停止したままにするかを制御します。
- 最大失敗タスク数 - ジョブが失敗としてマークされるまでに失敗する可能性があるタスクの最大数。
- タスクあたりの最大再試行回数 - 失敗したタスクが再試行される回数。
- 最大ワーカー数 - このジョブで同時に作業できるワーカーの最大数。
- Conda パッケージ - 3ds Max は conda を使用しないため、この設定は空である必要があります。
- Conda チャンネル - 3ds Max は conda を使用しないため、この設定は空である必要があります。

3ds Max 固有の設定



次の設定は 3ds Max レンダリングに固有です。

- プロジェクトパス - 3ds Max プロジェクトパス (自動検出)。
- 出力パス - レンダリングされたイメージが保存されるディレクトリ。

- Output Filename - レンダリングされたイメージファイルのベース名。### を使用してフレーム番号を表します。
- 出力ファイル拡張子 - レンダリングされたイメージのファイル形式 (.exr、.png、.jpg など)。
- 状態セット - レンダリングに使用する 3ds Max 状態セットを選択します。
- レンダラー - 3ds Max レンダリング設定の現在のレンダラー (読み取り専用)。
- ステレオカメラの選択 - ステレオプラグインが使用可能な場合は、ステレオカメラのレンダリングオプションを選択します。
- レンダリングするカメラ - 特定のカメラを選択するか、すべてのカメラをレンダリングします。
- フレーム範囲の上書き - オプションで、シーンのフレーム範囲をカスタム値で上書きします。

シーンの微調整

以下のオプションは、送信中にシーンを変更します。

- オブジェクト XRefs マージ - 外部オブジェクト参照をシーンにマージします。
- Merge Scene XRefs - 外部シーン参照をシーンにマージします。
- 送信されたファイルでマテリアルエディタをクリアする - マテリアルエディタからマテリアルを削除します。
- マテリアルエディタレンダラーのロック解除 - マテリアルエディタレンダラーのロックを解除します。
- シーンにカスタムマテリアルを適用する - すべてのシーンオブジェクトにカスタムマテリアルを適用します。

レンダリング要素

3ds Max のレンダリング要素は特殊な出力パスです。レンダリングされたイメージのさまざまな側面を個々のコンポーネントに分割して、高度な合成ワークフローとポストプロダクションワークフローを行います。これらの要素により、アーティストは拡散色、鏡面ハイライト、シャドウ、リフレクション、マテリアルプロパティなどの特定のレンダリングコンポーネントを分離できます。アーティストは、シーン全体を再レンダリングすることなく、ポストプロダクションでこれらのコンポーネントを正確に制御および調整できます。Deadline Cloud for 3ds Max は、高度なパス管理、V-Ray 統合、レンダリング中の自動設定による包括的なレンダリング要素サポートを提供します。

送信者は、以下のオプションで拡張レンダリング要素のサポートを提供します。

- レンダリング要素の変更 - このシーンのレンダリング要素設定の変更を有効にします。選択すると、レンダリング時に次のオプションが適用されます。
- 出力レンダリング要素 - レンダリング要素の出力の有効化/無効化を制御します。
- レンダリング要素パスの更新 - 送信中に出力パスを自動的に更新します。
- パスに名前/タイプを含める - 出力ディレクトリパスにレンダリング要素の名前またはタイプを追加します。
- ファイル名に名前/タイプを含める - 出力ファイル名にレンダリング要素の名前またはタイプを追加します。
- V-Ray 固有の設定 - V-Ray レンダリング要素の VFB 制御と分割バッファのサポート。
- 名前でレンダリング要素を無視する - 出力から特定のレンダリング要素を除外します。

他の送信者タブの詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

既知の制限事項

ジョブあたりの状態セット/バッチビューの最大数

[Open Job Description \(OpenJD\) 仕様](#)は、ジョブを最大 50 個のジョブパラメータに制限します。送信者は状態セットまたはバッチビューごとにステップごとのパラメータを作成するため、これにより、1 回のジョブ送信に含めることができる数に実質的な上限が設定されます。

送信者は、固定のグローバルパラメータと、状態セットまたはバッチビューの数に合わせてスケールするステップごとのパラメータを使用します。

パラメータグループ	カウント
基本パラメータ (シーンファイル、エラーチェック)	2
カメラパラメータ (特定のカメラが選択されている場合)	0 または 1
レンダリング要素パラメータ (シーンにレンダリング要素がある場合)	最大 10
デフォルトモードで設定された状態ごと (フレーム、出力path/name/format、解像度)	各 6

パラメータグループ	カウント
バッチレンダリングモードのバッチビューごと (フレーム、出力path/name/format、解像度、カメラ、シーンの状態、プリセット、ピクセルアスペクト)	各 10

実際の制限は次のとおりです。

送信モード	レンダリング要素	特定のカメラ	ジョブあたりの最大数
デフォルト	いいえ	いいえ	8 状態セット
デフォルト	いいえ	はい	7 つの状態セット
デフォルト	はい	いいえ	6 つの状態セット
デフォルト	はい	はい	6 つの状態セット
バッチレンダリング	いいえ	該当なし	4 つのバッチビュー
バッチレンダリング	はい	該当なし	3 つのバッチビュー

50 パラメータを超えるジョブの送信は、検証エラーで失敗します。制限よりも多くの状態セットまたはバッチビューをレンダリングする必要がある場合は、複数のジョブ送信に分割します。

V-Ray スタンドアロンタイルレンダリング

上級 V-Ray ユーザーの場合、3ds Max 内で V-Ray シーンファイル (.vrscene) をローカルにエクスポートし、タイルレンダリングをサポートするスタンドアロンジョブバンドルとして送信できます。このワークフローは、タイル化によってメモリフットプリントを削減し、レンダリング時間を最適化できる高解像度レンダリングに特に役立ちます。

このワークフローを使用するタイミング

Linux ワーカーでの V-Ray Standalone を使用したタイルレンダリングは、以下に役立ちます。

- 高解像度レンダリング (屋外広告、高解像度エンターテインメントコンテンツ)。

- 小さなリージョンを処理することでメリットが得られる、メモリ要件が高いシーン。
- 画像を並行してレンダリングされた均等なサイズの領域に分割することで、レンダリングリソースを最適化します。
- 並列処理によるレンダリング時間の最小化。
- Windows ワーカーの代わりに Linux Amazon Elastic Compute Cloud (Amazon EC2) ワーカーを使用することで、インフラストラクチャコストを削減します (Linux Amazon EC2 インスタンスは通常、同等の Windows インスタンスよりも 1 時間あたりのレートが低くなります)。

V-Ray シーンファイルのエクスポート

V-Ray for 3ds Max には、V-Ray Standalone でレンダリングできるすべてのシーン情報 (ジオメトリ、ライト、シェーダー) を含む .vrscene ファイルを作成する Scene Exporter が含まれています。

V-Ray シーンファイルのエクスポートするには:

1. 3ds Max で、必要に応じて V-Ray レンダリング設定を構成します。
2. V-Ray Scene Exporter を使用して、シーンを .vrscene ファイルとしてエクスポートします。

エクスポートされたファイルは、完全なシーンデータを含むテキストベースの形式です。

タイルレンダリングジョブの送信

.vrscene ファイルをエクスポートしたら、スタンドアロンタイルレンダリングジョブバンドルを使用して、最適化されたレンダリングジョブを Deadline Cloud に送信できます。

ジョブバンドルの作成と送信に関する一般的な情報については、[「Deadline Cloud Developer Guide」の「Open Job Description templates for Deadline Cloud」](#)を参照してください。

リファレンス実装:

deadline-cloud-samples リポジトリの [tile_render_with_vray_linux](#) サンプルは、次のことを示しています。

- 大きなイメージをタイルに分割する方法。
- Linux ワーカーでのタイルの並列レンダリング。
- レンダリングが完了した後の自動タイルアセンブリ。

このジョブバンドルは、Deadline Cloud CLI を使用して送信できます。

```
deadline bundle submit <path-to-job-bundle>
```

または、GUI 送信者を使用します。

```
deadline bundle gui-submit <path-to-job-bundle>
```

このアプローチの利点:

- レンダリングタスクあたりのメモリ使用量を削減しました。
- タイルの並列処理により、全体的なレンダリング時間が短縮されます。
- Deadline Cloud ファーム全体のリソース使用率が向上します。
- シーンの要件に基づいてタイルのディメンションをカスタマイズする柔軟性。
- Windows ワーカーの代わりに Linux ワーカーを使用することによるコスト削減 (Linux Amazon EC2 インスタンスは通常、同等の Windows インスタンスよりもコストが低くなります)。

ジョブバンドル構造

タイルレンダリングジョブバンドルでは、Open Job Description テンプレートを使用して以下を定義します。

- 水平タイルと垂直タイルの数を指定するためのジョブパラメータ。
- 各タイルに個々のタスクを作成するタスクパラメータ。
- 各タイルを並列処理するレンダリングステップ。
- レンダリングの完了後にタイルを結合するアセンブリステップ。

要件

- V-Ray for 3ds Max with Scene Exporter。
- Linux ワーカーで構成された Deadline Cloud ファーム。
- ワーカーノードにインストールされた V-Ray Standalone。
- タイルアセンブリ用の FFmpeg または同様のツール (conda を使用して提供できます)。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアのバージョンのみをサポートおよびテストします。アーティストが使用する 3ds Max のバージョンが、フリートのホスト設定で設定された 3ds Max のバージョンと互換性があることを確認する必要があります。

ホスト設定スクリプトを使用して、古い 3ds Max バージョンをサポートできます。ただし、古い Python バージョンが原因で、統合された送信者が機能しない場合があります。このような場合でも、カスタムジョブバンドルは Deadline Cloud ジョブとして送信できます。

3ds Max レンダラー

Deadline Cloud は、3ds Max ジョブを含むホスト設定スクリプトを使用する場合、次のレンダラーを使用した 3ds Max ジョブのレンダリングをサポートします。

レンダラー	レンダラーのバージョン	ホスト設定スクリプトの提供	使用状況ベースのライセンスサポート
Autodesk スキャンライン	組み込み	該当なし	該当なし
Autodesk Raytracer (ART)	組み込み	該当なし	該当なし
カオス V-Ray 6	6.x	はい	はい
カオス V-Ray 7	7.x	はい	はい
コロナ	最新	はい	いいえ

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [3ds Max 送信者とアダプター](#)
- [Deadline Cloud サンプル \(3ds Max ワークフローの例用\)](#)
- [3ds Max Host Config の例](#)

Autodesk Maya

Autodesk Maya は、ビデオゲーム、アニメーション映画、テレビシリーズ、視覚効果など、インタラクティブな 3D アプリケーションの作成に使用される 3D コンピュータアニメーション、モデリング、シミュレーション、レンダリングソフトウェアです。Maya は Deadline Cloud で完全にサポートされており、送信者、conda パッケージ、使用ベースのライセンス、レンダリングパフォーマンスを向上させるアダプターなどの包括的な統合が可能です。このガイドでは、Autodesk Maya で Deadline Cloud を使用して、レンダリングタスクを複数のマシンに分散することでプロジェクトをより迅速にレンダリングするための step-by-step の手順を示します。

サポートの概要

Maya は、次のコンポーネントでサポートされています。

- 送信者: Maya から直接ジョブを送信するための統合プラグイン。
- Conda パッケージ: 送信者を使用する場合のサービスマネージドフリートへの自動インストール。
- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows、macOS、Linux の送信者サポートと Windows と Linux のワーカーサポート。
- 使用状況ベースのライセンス: Maya およびレンダラーライセンスの Pay-as-you-go。

Maya バージョンの互換性

次の表は、Maya バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート	エンジンのレンダリング	使用状況ベースのライセンス
2024	Windows、macOS、Linux	Linux	Maya Software、Arnold (MtoA)	使用状況ベースのライセンスが利用可能
2025	Windows、macOS、Linux	Linux	Maya Software、Arnold	使用状況ベースのライセンスが利用可能

メジャーバージョン	送信者のサポート	Conda サポート	エンジンのレンダリング	使用状況ベースのライセンス
			(MtoA)、V-Ray、Redshift	
2026	Windows、macOS、Linux	Linux	Maya Software、Arnold (MtoA)、V-Ray、Redshift	使用状況ベースのライセンスが利用可能

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルのサービスマネージドフリートで使用できる Maya に適用されるすべての conda パッケージを示します。

OS	パッケージ	バージョン	注意事項
Linux	マヤ	2024	Maya ソフトウェアレンダラーを含む
Linux	マヤ	2025	Maya ソフトウェアレンダラーを含む
Linux	マヤ	2026	Maya ソフトウェアレンダラーを含む
Linux	Maya-mtoa	2024.5.3	Arnold for Maya 2024
Linux	Maya-mtoa	2025.5.4	Arnold for Maya 2025
Linux	Maya-mtoa	2026.5.5	Arnold for Maya 2026
Linux	Maya-openjd		Maya アダプターを含む
Linux	maya-redshift	2025.4	Redshift for Maya 2025

OS	パッケージ	バージョン	注意事項
Linux	maya-redshift	2026.2.1	Redshift for Maya 2026
Linux	Maya-vray	2025.7	V-Ray for Maya 2025
Linux	Maya-vray	2026.7	V-Ray for Maya 2026

開始方法

Deadline Cloud で Maya を使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。キューは、`deadline-cloud conda` チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。
2. Deadline Cloud Submitter を使用してアーティストワークステーションに Deadline Cloud モニターと Maya 送信者をインストールし、インストーラをモニタリングします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. 統合された送信者を使用して Maya から直接ジョブをキューに送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

インストール

Autodesk Maya の Deadline Cloud 送信者をインストールするには、次の環境を準備します。

- Windows、Linux、または macOS ワークステーション。
- Autodesk Maya 2024、2025、または 2026 のインストール。
- オプション: Arnold (MtoA 5.3.5 以降)、V-Ray、または Redshift for Maya のインストール。
- [Deadline Cloud Monitor](#) がインストールされました。
- 次のいずれかを使用して Deadline Cloud ファームにアクセスします。
 - サービスマネージドフリート。
 - Autodesk Maya とライセンス設定によるカスタマーマネージドフリート。

送信者のインストール

Autodesk Maya 送信者拡張機能を使用すると、Maya 内から直接 Deadline Cloud にジョブを送信できます。送信者をインストールするには:

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラを実行し、画面の指示に従います。
3. インストール後に Maya を起動します。

送信者の更新

送信者を最新バージョンに更新するには、最新の送信者インストーラをダウンロードして実行します。

Autodesk Maya 送信者の使用

Maya の Deadline Cloud 送信者を使用するには、ファームに Maya 対応フリートが設定されていること、および送信者がインストールされていることを確認します。インストール手順については、「」を参照してください[インストール](#)。Deadline Cloud にアクセスするには、Deadline Cloud モニターにログインするか、設定プロファイルを介して AWS 認証情報を指定します。

ジョブを送信する

Maya から Deadline Cloud にジョブを送信するには

1. Maya ファイルを保存します。
2. Maya のシェルフで、Deadline Cloud ボタンを選択します。
3. ダイアログのタブを使用してジョブをカスタマイズします。
4. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
5. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

共有ジョブ設定

The screenshot shows a window titled "Submit to AWS Deadline Cloud" with four tabs: "Shared job settings", "Job-specific settings", "Job attachments", and "Host requirements". The "Shared job settings" tab is active, displaying the following configuration:

Job Properties

- Name: `Sophie_user-guide.ma`
- Description: (empty)
- Priority: `50`
- Initial state: `READY`
- Maximum failed tasks count: `20`
- Maximum retries per task: `5`
- Maximum worker count: No max worker count, Set max worker count

Deadline Cloud settings

- Farm: `conda-release-testing-shared-farm-LHR`
- Queue: `conda-release-testing-queue-linux`

Queue Environment: Conda

- Conda Packages: `maya=2026.* maya-openjd=0.15.* maya-mtoa`
- Conda Channels: `deadline-cloud`

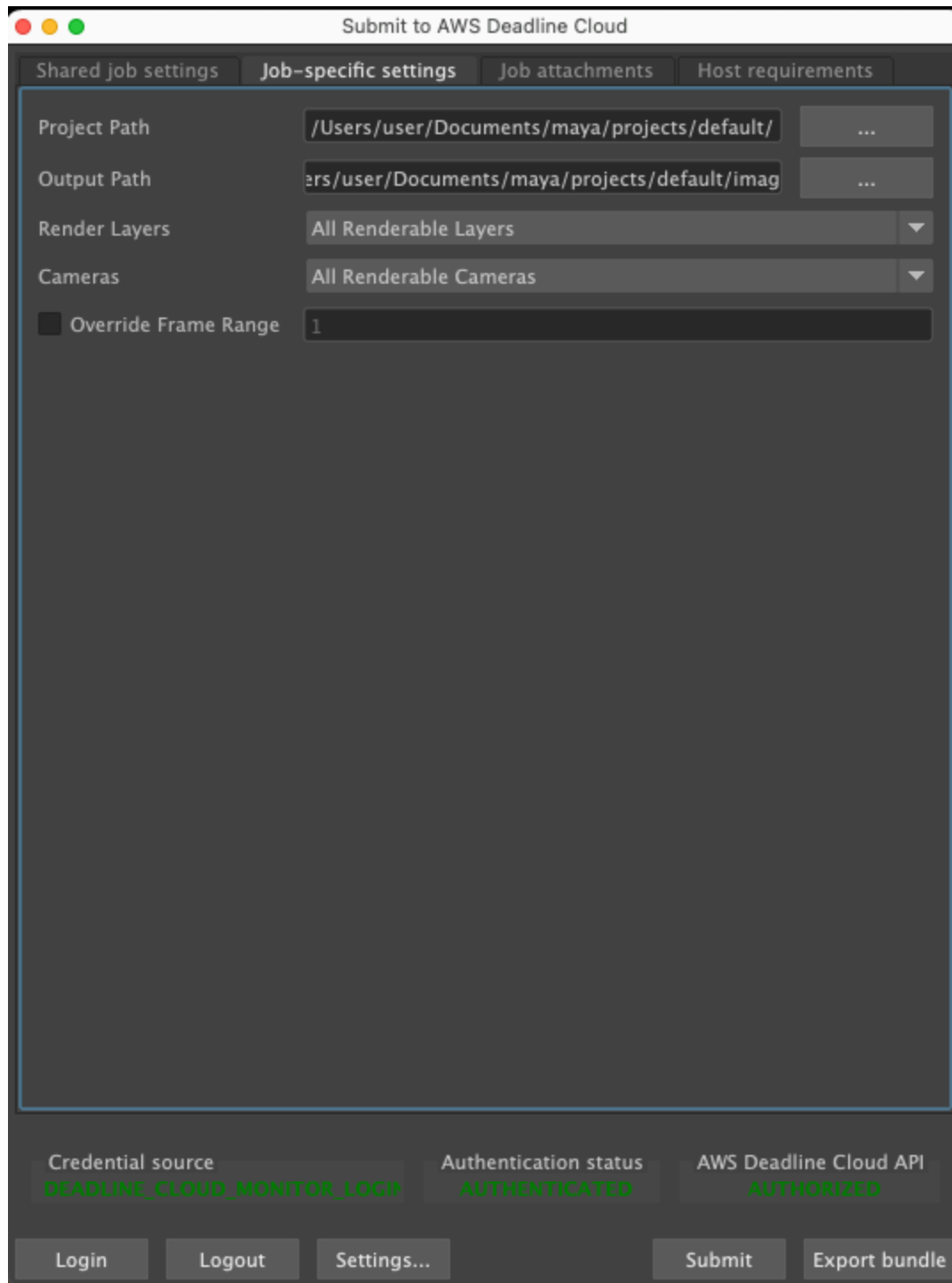
At the bottom, there are three status indicators: "Credential source" (DEADLINE_CLOUD_MONITOR_LOGIN), "Authentication status" (AUTHENTICATED), and "AWS Deadline Cloud API" (AUTHORIZED). Below these are buttons for "Login", "Logout", "Settings...", "Submit", and "Export bundle".

ジョブ全体に適用される設定:

- ファームの選択 - ジョブがレンダリングするファームを選択します。
- キューの選択 - 選択したファーム内の特定のキューを選択します。

- ジョブ名 - レンダリングジョブにわかりやすい名前を付けます。
- ジョブの説明 - レンダリングジョブに関するオプションの詳細を追加します。
- Priority - キュー管理のジョブ優先度を設定します。
- 初期状態 - ジョブをすぐに開始するか、一時停止したままにするかを制御します。
- 最大失敗タスク数 - ジョブが失敗としてマークされるまでに失敗する可能性があるタスクの最大数を設定します。
- タスクあたりの最大再試行回数 - 失敗したタスクが再試行される回数を設定します。
- 最大ワーカー数 - このジョブで同時に作業できるワーカーの最大数を設定します。
- Conda パッケージ - レンダリングに必要な追加の conda パッケージを指定します。
- Conda Channels - パッケージのインストール用にカスタム conda チャンネルを定義します。

Maya 固有の設定



Maya レンダリングに固有の設定:

- プロジェクトパス - Maya プロジェクトパスを指定します (自動検出)。
- 出力パス - レンダリングされたイメージが保存されるディレクトリを指定します。

- Output Filename - レンダリングされたイメージファイルのベース名を入力します。
- レンダラー - 使用するレンダラー (Arnold、V-Ray、Redshift、または Maya ソフトウェア) を選択します。
- レンダリングするカメラ - 特定のカメラを選択するか、すべてのレンダリング可能なカメラをレンダリングします。
- フレーム範囲の上書き - オプションで、シーンのフレーム範囲をカスタム値で上書きします。
- レンダリングレイヤー - レンダリングするレンダリングレイヤーを選択します。

オプションのタブ

送信中にシーンを変更するオプション:

- ジョブアタッチメント (オプション) - ジョブにアップロードしてアタッチするファイルを選択します。ファイルはデフォルトで自動的に検出され、アタッチされます。
- ホスト要件 (オプション) - このジョブのタスクのピックアップの対象となるホストのタイプを指定できます。

送信者タブの詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとしています。ワークステーションバージョンの Maya が上記のバージョン表に表示されない場合、これは失敗します。

サポートされていないバージョンの Maya が必要な場合は、次のオプションがあります。

- Maya からジョブを送信するときは、CondaPackages キューパラメータを上書きして、ワーカーで使用するサポートされているバージョンを指定できます (例: maya=2026, maya-openjd=*)。これは、シーンで使用される機能や、Maya がワークステーションバージョンのシーンとどのように連携するかに応じて、機能しない場合があります。
- ワーカーにインストールする目的のバージョンのカスタム conda レシピとチャンネルを構築できます。サポートされているバージョンの conda レシピを出発点として使用します。

- [Maya conda レシピ](#)
- [Maya OpenJD アダプター conda レシピ](#)

カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

Maya レンダリングエンジン

Maya は、Deadline Cloud と完全に互換性がある複数のレンダーエンジンをサポートしています。

レンダリングエンジン	説明	GPU サポート	注意事項	使用状況ベースのライセンス
Maya ソフトウェア	組み込み CPU レンダラー	CPU ベース	基本的な機能を備えたレガシーレンダラー	Maya に付属
アーノルド (MtoA)	モンテカルロレイトレーサー	GPU/CPU ハイブリッド	本番稼働用品質レンダリング、MtoA 5.3.5 以降が必要	2024-2026 で利用可能
V-Ray	サードパーティーのフォトリアリスティックレンダラー	GPU/CPU ハイブリッド	個別のライセンスが必要	2025-2026 で利用可能
Redshift	GPU アクセラレーションレンダラー	GPU 最適化	個別のライセンスが必要	2025-2026 で利用可能

すべてのレンダリングエンジンは、Maya 統合送信者によって自動的に検出および設定されます。送信者は、適切な依存関係処理とシーンファイル管理を維持します。

Maya プラグイン

プラグイン	プラグインのバージョン	Conda レシピの提供	提供されている SMF Conda パッケージ	使用状況ベースのライセンスサポート
アーノルド (MtoA)	2024.5.3、 2025.5.4、 2026.5.5	はい	はい	はい
V-Ray	2025.7、2026.7	はい	はい	はい
Redshift	2025.4、20 26.2.1	はい	はい	はい

Arnold for Maya (MtoA)

Arnold は maya-mtoa conda パッケージを使用してサポートされており、Maya 統合送信者を使用するときに自動的にインストールされます。レンダリングに Arnold を使用する場合、追加のライセンス料金が適用されます。

Conda レシピ: [maya-mtoa conda レシピ](#)

V-Ray プラグイン

V-Ray は maya-vray conda パッケージを使用してサポートされており、Maya 統合送信者を使用するときに自動的にインストールされます。レンダリングに V-Ray を使用する場合、追加のライセンス料金が適用されます。

Conda レシピ: [maya-vray conda レシピ](#)

Redshift プラグイン

Redshift は maya-redshift conda パッケージを使用してサポートされており、Maya 統合送信者を使用して自動的にインストールされます。レンダリングに Redshift を使用する場合、追加のライセンス料金が適用されます。

Conda レシピ: [maya-redshift conda レシピ](#)

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [Maya 送信者のソースコード](#)
- [Maya conda レシピ](#)

Autodesk VRED

Autodesk VRED は、複雑な 3D データをリアルな仮想環境で実現するプロフェッショナルな 3D ビジュアライゼーションおよび仮想プロトタイプソフトウェアです。VRED は、特に自動車業界で、製品プレゼンテーション、設計レビュー、仮想プロトタイプを作成するためにデザイナーやエンジニアによって広く使用されています。このガイドを使用して、複数のマシンにタスクを分散して Deadline Cloud と Autodesk VRED でレンダリングします。

サポートの概要

VRED は、以下のコンポーネントを使用して Deadline Cloud によって部分的にサポートされています。

- 送信者: シーンとアセットの自動検出機能を備えた VRED Pro から直接ジョブを送信するための統合送信者。
- Conda パッケージ: vredcore パッケージを使用した Linux ワーカー向けのサービスマネージドフリートへの自動インストール。
- クロスプラットフォーム互換性: 自動パスマッピングによる Linux のワーカーサポートを備えた Windows の送信者サポート。(VRED Conda パッケージは Linux でのみ使用できます。Windows ワーカーは手動でインストールする必要があります)。
- BYOL ライセンス: VRED には Bring Your Own License (BYOL) が必要です。Deadline Cloud の他の DCC アプリケーションとは異なり、使用状況ベースのライセンスは VRED では使用できません。レンダーファームフリートで使用できる有効な VRED ライセンスがあり、ワーカーからアクセスできるようにライセンスサーバーを設定する必要があります。

VRED バージョンの互換性

次の表は、VRED バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート	使用状況ベースのライセンス
2026	Windows	Linux	BYOL が必要
2025	Windows	Linux	BYOL が必要

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルのサービスマネージドフリートで使用できる VRED に適用されるすべての conda パッケージを示します。

OS	パッケージ	バージョン	注意事項
Linux	vredcore	2025	Linux 用 VRED Core
Linux	vredcore	2026	Linux 用 VRED Core

要件

Deadline Cloud で VRED を使用するには、以下が必要です。

- 有効なライセンスを持つ VRED Pro または VRED Core 2025/2026
- Python 3.11 以降
- NVIDIA GPU ドライバー 553.xx (最適なパフォーマンスのために推奨)
- レンダーファームフリートからアクセスできる有効な VRED ライセンス
- オプション: レイトレーシングでリージョンレンダリングを使用する場合のタイルアセンブリの ImageMagick 静的バイナリ

Important

VRED 統合には、独自のライセンス (BYOL) が必要です。レンダーファームフリートで使用できる有効な VRED ライセンスがあり、ワーカーノードからアクセスできるようにライセン

サーバーを設定する必要があります。詳細については、[「サービスマネージドフリートのカスタムライセンスサーバーに接続する」](#)を参照してください。

開始方法

Deadline Cloud で VRED を使用するには:

1. サーマネージドフリートを作成し、キューに関連付けます。フリートが VRED ライセンスサーバーにアクセスできることを確認します。
2. Deadline Cloud Submitter とモニターインストーラを使用して、アーティストワークステーションに Deadline Cloud モニターと VRED 送信者をインストールします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. VRED を開き、シーンファイルをロードします。
4. 統合送信者を使用して VRED から直接ジョブを送信するには、メニューから Deadline Cloud > Submit to Deadline Cloud を選択します。
5. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

インストール

Autodesk VRED 用の Deadline Cloud 送信者をインストールするには、次の環境を準備します。

- Windows 10+ ワークステーション。
- VRED Pro 2025 または 2026 のインストール。
- Python 3.11 以降。
- 次のいずれかを使用して Deadline Cloud ファームにアクセスします。
 - VRED ソフトウェアとライセンスが設定されているサービスマネージドフリート。
 - VRED とライセンスが設定されているカスタマーネージドフリート。

Important

Deadline Cloud for VRED には、独自のライセンス (BYOL) が必要です。レンダーファームフリートで使用できる有効な VRED ライセンスが必要です。

送信者のインストール

Autodesk VRED 送信者プラグインを使用すると、VRED 内から直接 Deadline Cloud にジョブを送信できます。送信者をインストールするには:

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラを実行し、画面の指示に従います。
3. VRED Pro を開きます。
4. **編集 > 設定** を選択します。
5. 設定ウィンドウで、**全般設定**を選択し、**スクリプト**を選択します。
6. Python サンドボックスの有効化をクリアします。
7. スクリプトセクションで、次のテキストをセクションの最後に追加します。

```
from DeadlineCloudForVRED import DeadlineCloudForVRED
DeadlineCloudForVRED()
```

8. **[保存]** を選択します。
9. VRED Pro を再起動します。VRED が開くと、Deadline Cloud メニューがメニューバーに表示されます。

手動インストールまたは開発者ワークフローについては、[GitHub リポジトリの手動インストール手順](#)を参照してください。

送信者の更新

送信者を最新バージョンに更新するには、最新の送信者インストーラをダウンロードして実行します。

Autodesk VRED 送信者の使用

VRED に Deadline Cloud 送信者を使用する前に、ファームに VRED 対応フリートが設定されており、送信者がインストールされていることを確認してください。また、Deadline Cloud モニターで認証するか、設定プロファイルを介して AWS 認証情報を指定する必要があります。

ジョブの送信

VRED から Deadline Cloud にジョブを送信するには

1. VRED シーンファイルを保存します。
2. VRED のメニューバーで、Deadline Cloud > Submit to Deadline Cloud を選択します。
3. Deadline Cloud でまだ認証していない場合は、ログインを選択し、表示されるブラウザウィンドウで認証情報を使用して認証します。
4. ダイアログのタブを使用してジョブをカスタマイズします。
5. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
6. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

共有ジョブ設定

The screenshot shows a dark-themed window titled "Submit to AWS Deadline Cloud" with a close button (X) in the top right corner. The window is divided into four tabs: "Shared job settings" (selected), "Job-specific settings", "Job attachments", and "Host requirements".

Job Properties

- Name:
- Description:
- Priority: (with left and right arrow icons)
- Initial state: (with a dropdown arrow)
- Maximum failed tasks count: (with left and right arrow icons)
- Maximum retries per task: (with left and right arrow icons)
- Maximum worker count: No max worker count, Set max worker count

Deadline Cloud settings

- Farm: testing-farm-PDX-DEV
- Queue: testing-queue-PDX-DEV

Queue Environment: BYOL License Forwarding

- LicenseInstanceid:
- LicenseInstanceRegion:
- LicensePorts:

Queue Environment: Conda

- Conda Packages:
- Conda Channels:

At the bottom left, there is a green checkmark icon and a dropdown menu showing a redacted name followed by "-us-west-2".

At the bottom right, there are three buttons: "Settings...", "Submit", and "Export bundle".

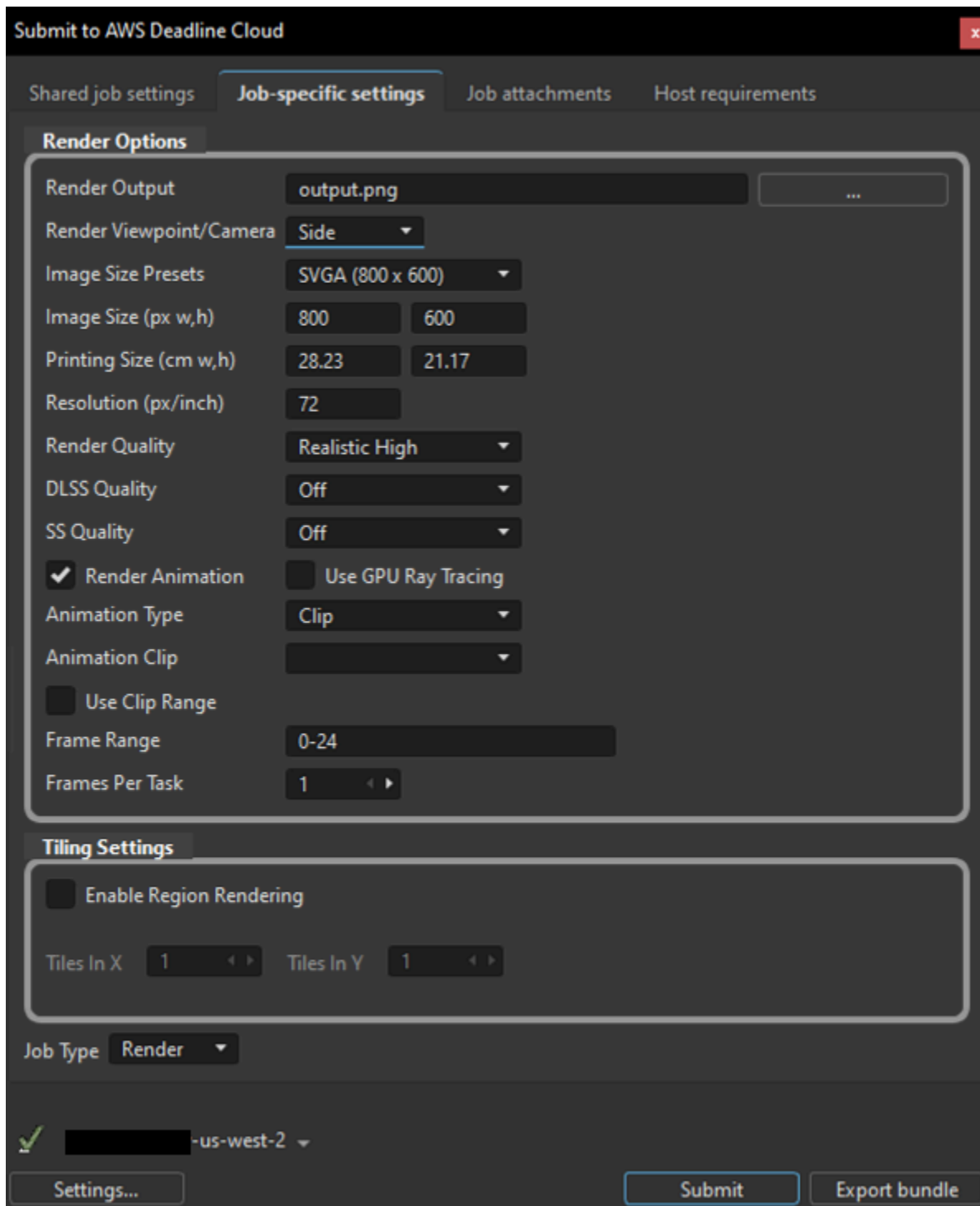
これらの設定はジョブ全体に適用されます。

- 名前 - レンダリングジョブのわかりやすい名前。
- 説明 - レンダリングジョブに関するオプションの詳細。

- Priority - キュー管理のジョブ優先度 (デフォルト: 50)。
- 初期状態 - ジョブがすぐに開始される (READY) か、一時停止されたままになるか。
- 最大失敗タスク数 - ジョブが失敗としてマークされる前に失敗する可能性がある最大タスク数 (デフォルト: 20)。
- タスクあたりの最大再試行回数 - 失敗したタスクが再試行される回数 (デフォルト: 5)。
- 最大ワーカー数 - このジョブを同時に処理できる最大ワーカー数。
- Farm - ジョブがレンダリングされるファーム。
- キュー - 選択したファーム内の特定のキュー。
- Conda パッケージ - ジョブ環境の Conda パッケージ (VRED バージョン用に自動設定)。
- Conda チャンネル - パッケージ解決用の Conda チャンネル。

VRED 固有の設定

ジョブ固有の設定タブには、VRED に固有のレンダリングオプションが含まれています。



レンダリングオプション

- レンダリング出力 - レンダリングされたイメージの出力ファイルパスとベースファイル名。参照ボタン (...) を使用して出力ディレクトリを選択し、ファイル名を指定します。

- レンダリングビューポイント/カメラ - レンダリング元のビューポイントまたはカメラを選択します。ドロップダウンには、シーンで使用できるすべてのビューポイントが一覧表示されます。
- イメージサイズプリセット - 一般的なイメージディメンション (S" 800x600、HD 1080、4K など) の迅速な選択。
- イメージサイズ (px w,h) - レンダリングされた出力の幅と高さ。これらの値は、プリセットを選択すると自動的に更新されます。または、カスタムディメンションを入力することもできます。
- 印刷サイズ (cm w,h) - センチメートル単位の物理的な印刷寸法。この値は、イメージのサイズと解像度にリンクされています。
- 解像度 (px/inch) - Dots-per-inch (DPI) 設定。
- レンダリング品質 - レンダリング用の品質プリセット。オプションには以下が含まれます。
 - 分析の低/高
 - リアルな低/高
 - レイトレース
 - 非フォトリアリスティック (NPR)
- DLSS 品質 - NVIDIA Deep Learning Super Sampling 品質レベル (オフ、パフォーマンス、バランス、品質、超パフォーマンス)。互換性のある NVIDIA GPU が必要です。
- SS 品質 - スーパーサンプリングのアンチエイリアシング品質 (オフ、低、中、高、超高)。DLSS を有効にすると、この設定が上書きされます。
- アニメーションのレンダリング - 有効にすると、単一のフレームではなくアニメーションシーケンスをレンダリングします。追加のアニメーションオプションを表示します。
- GPU レイトレーシングを使用する - GPU アクセラレーションレイトレーシングを有効にして、高品質のレンダリングを実現します。互換性のある GPU ハードウェアが必要です。
- アニメーションタイプ - レンダリングするアニメーションのタイプ (クリップまたはタイムライン)。レンダリングアニメーションが有効になっている場合に表示されます。
- アニメーションクリップ - ドロップダウンからレンダリングするアニメーションクリップを選択します。Animation Type が Clip に設定されている場合に表示されます。
- クリップ範囲を使用する - 有効にすると、選択したアニメーションクリップで定義されたフレーム範囲が使用されます。
- フレーム範囲 - レンダリングする開始フレームと終了フレーム (例: 0-24)。カスタム範囲を指定するか、クリップの範囲を使用できます。
- タスクあたりのフレーム - 各ワーカーがタスクごとにレンダリングする連続フレームの数 (デフォルト: 1)。値を大きくすると、タスクの初期化オーバーヘッドが削減され、レンダリング効率が向上しますが、進行状況の追跡の粒度は低くなります。たとえば、タスクごとのフレームを 5 に設

定すると、タスク 1 はフレーム 1~5 をレンダリングし、タスク 2 はフレーム 6~10 をレンダリングします。

タイル設定

リージョンレンダリング (タイル化) は、各フレームを個別のタスクとして個別にレンダリングする小さなタイルに分割し、それらを最終的なイメージにアSEMBルします。タイル化は、大規模で複雑なレンダリングのパフォーマンスを向上させることができます。

- リージョンレンダリングを有効にする - タイルベースのレンダリングを有効にします。有効にすると、GPU Ray トレースの使用が自動的に有効になります。
- X のタイル - 水平タイル除算の数 (デフォルト: 1)。
- Y のタイル - 垂直タイル除算の数 (デフォルト: 1)。

Important

リージョンのレンダリングでは、GPU Ray トレースの使用を有効にする必要があります。シーンファイルは、正しい出力を生成するためにレイトレーシング (十分な照明があるなど) 用に適切に設定する必要があります。レイトレーシング用に設定されていないシーンでリージョンレンダリングを使用すると、黒いタイルが生成されます。

タイルアセンブリの要件: リージョンレンダリングでは、タイルを最終イメージにアSEMBルするために ImageMagick をレンダリング環境で使用できるようにする必要があります。サービスマネージドフリートで ImageMagick を提供する 1 つの方法は、(conda-forge チャンネルからなど) `imagemagick conda` パッケージをキュー環境に追加することです。conda パッケージの設定手順の詳細については、[「キュー環境を使用したジョブの設定」](#)を参照してください。

ジョブタイプ

- ジョブタイプ - 送信するジョブのタイプを選択します。
 - レンダリング - シーンからイメージまたはアニメーションフレームをレンダリングします (デフォルト)。
 - Sequencer - シーンで定義された VRED Sequencer ワークフローを実行します。

他の送信者タブ (ジョブアタッチメント、ホスト要件) の詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとしています。これは、ワークステーションバージョンの VRED が上記のバージョンテーブルに表示されない場合に失敗します。

サポートされていないバージョンの VRED が必要な場合は、ワーカーにインストールする目的のバージョンのカスタム Conda レシピとチャンネルを構築できます。以下でリンクされているサポートされているバージョンの Conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム Conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

VRED レンダリングエンジン

VRED は Deadline Cloud ジョブで次のレンダリングエンジンをサポートしています。

レンダリングエンジン	説明	GPU サポート	注意事項
OpenGL	リアルタイムレンダラー	GPU 最適化	インタラクティブビジュアライゼーション
レイトレース	高品質のレンダラー	GPU/CPU ハイブリッド	本番稼働品質のレンダリング

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [VRED 送信者とアダプター](#)

- [VRED Conda レシピ](#)は、サポートされているバージョンの GitHub で利用できます。

ブレンダー

Blender は、アニメーション映画、視覚効果、アート、3D プリントモデル、モーショングラフィックス、インタラクティブな 3D アプリケーション、バーチャルリアリティ、コンピュータゲームの作成に使用される無料のオープンソースの 3D コンピュータグラフィックスソフトウェアツールセットです。3D Blender AWS は Deadline Cloud (Deadline Cloud) でサポートされており、送信者、conda パッケージ、レンダリングパフォーマンスを向上させるアダプターなどの包括的な統合が可能です。このガイドでは、Blender で Deadline Cloud を使用して、レンダリングタスクを複数のマシンに分散することでプロジェクトをより迅速にレンダリングするための step-by-step の手順を示します。

サポートの概要

Blender は、次のコンポーネントでサポートされています。

- 送信者: Blender からジョブを直接送信するための統合送信者。シーンとアセットの自動検出機能を備えています。
- Conda パッケージ: サービスマネージドフリートへの自動インストールの Deadline Cloud。
- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows、macOS、Linux の送信者サポートと、自動パスマッピングによる Windows と Linux のワーカーサポート。

Blender バージョンの互換性

次の表は、Blender バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート	エンジンのレンダリング
3.6	Windows、macOS、Linux	Linux	サイクル、Eevee、Workbench
4.2	Windows、macOS、Linux	Linux	サイクル、Eevee、Workbench

メジャーバージョン	送信者のサポート	Conda サポート	エンジンのレンダリング
4.5	Windows、macOS、Linux	Linux	サイクル、Eevee、Workbench
5.0	Windows、macOS、Linux	Linux	サイクル、Eevee、Workbench
5.1	Windows、macOS、Linux	Linux	サイクル、Eevee、Workbench

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルでサービスマネージドフリートで使用できる Blender に適用されるすべての conda パッケージを示します。

OS	パッケージ	バージョン	注意事項
Linux	ブレンダー	3.6	すべての組み込みレンダーエンジンを含む
Linux	ブレンダー	4.2	すべての組み込みレンダーエンジンを含む
Linux	ブレンダー	4.5	すべての組み込みレンダーエンジンを含む
Linux	ブレンダー	5.0	すべての組み込みレンダーエンジンを含む
Linux	ブレンダー	5.1	すべての組み込みレンダーエンジンを含む

OS	パッケージ	バージョン	注意事項
Linux	ブレンダー-openjd		Blender Adaptor を含む

開始方法

Deadline Cloud で Blender を使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。キューは、期限クラウド conda チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。
2. Deadline Cloud モニターと送信者インストーラを使用して、アーティストワークステーションに Deadline Cloud モニターと Blender 送信者をインストールします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. 統合された送信者を使用して、Blender から直接ジョブをキューに送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

インストール

Deadline Cloud for Blender 送信者をインストールするには、以下が必要です。

- Windows、macOS、または Linux ワークステーション。
- Blender 3.6 以降。

Deadline Cloud for Blender 送信者をインストールするには、次の 3 つの方法があります。

- Deadline Cloud 送信者インストーラの使用 (推奨)。
- Blender から送信者をインストールします。
- [送信元から送信者を手動でインストールします。](#)

Deadline Cloud 送信者インストーラの使用

Deadline Cloud for Blender 送信者は、Deadline Cloud 送信者インストーラを使用してインストールできます。

送信者をインストールするには

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラーを実行します。
 - プロンプトが表示されたら、送信者を使用する Blender の各バージョンを選択します。
3. Blender を起動します。
4. 「Deadline Cloud に送信」オプションのレンダリングメニューを確認して、インストールを確認します。

アドオンがレンダリングメニューから使用できない場合は、手動で有効にする必要があります。

送信者アドオンを手動で有効にするには

1. 編集メニューで、設定... を選択します。
2. 左側のバーでファイルパスを選択します。
3. スクリプトディレクトリセクションを見つけ、+ を選択します。
4. [名前] にpythonと入力してください。
5. パス には、Blender 送信者インストールの python ディレクトリへのパスを入力します。
6. Blender を再起動して変更を有効にします。

Blender からの送信者のインストール

Note

これは実験的な機能であり、変更される可能性があります。

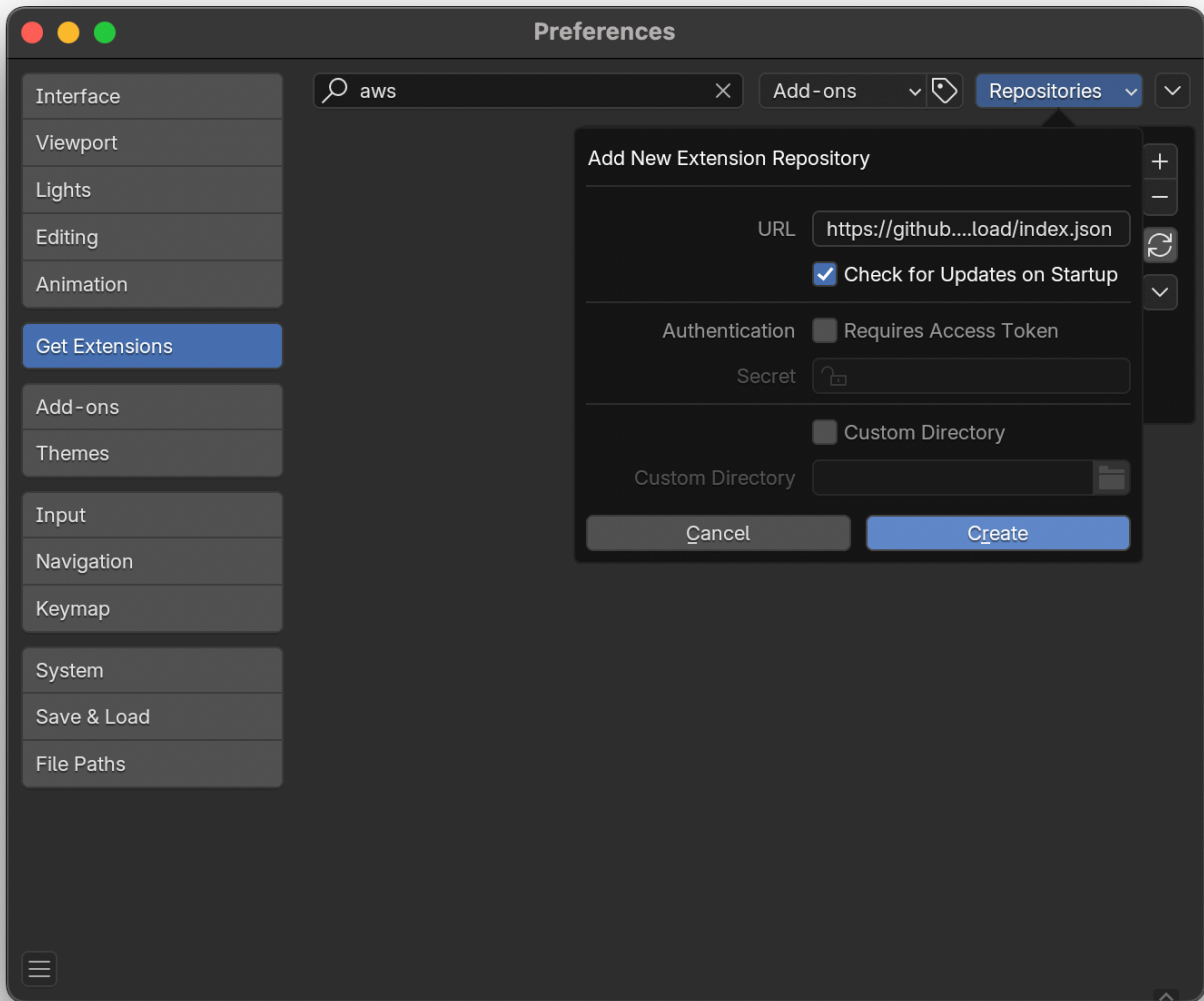
Blender の拡張機能を使用して、Blender 内から Blender 送信者をインストールおよび更新できます。

Blender 拡張機能を使用して Blender 送信者をインストールするには、以下が必要です。

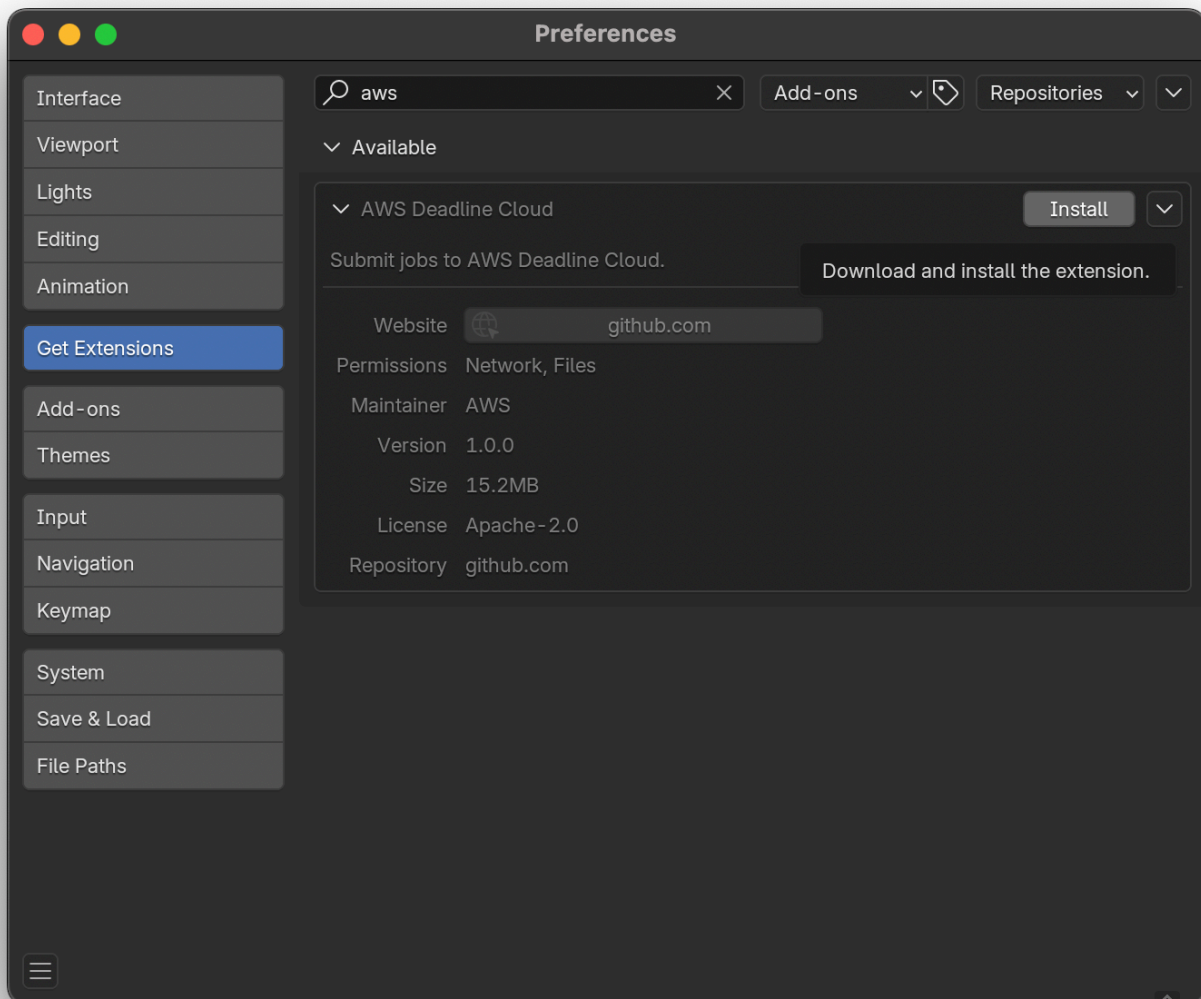
- Blender 4.2 以降。
- 一貫したインターネットアクセスを備えたワークステーション。

Blender 送信者を拡張機能として追加するには

1. Blender を開きます。
2. 編集メニューで、設定... を選択します。
3. 左側のバーで拡張の取得を選択します。
4. リポジトリ、+、リモートリポジトリの追加を選択します。



5. [URL] には、<https://github.com/aws-deadline/deadline-cloud-for-blender/releases/latest/download/index.json> と入力します。
6. 起動時に更新をチェックを選択し、作成を選択します。
7. 「利用可能」の「Deadline Cloud」エントリで、「インストール」を選択します。



これでアドオンがインストールされました。レンダリングメニューの新しい「Deadline Cloud に送信」オプションを使用できます。

更新が利用可能になると、拡張機能の取得セクションの Deadline Cloud エントリの横に更新ボタンが表示されます。

Blender 送信者の使用

Deadline Cloud for Blender 送信者を使用するには、以下が必要です。

- Deadline Cloud に送信するプロファイル。

- 送信先の Deadline Cloud ファームとキュー。

ジョブを送信する

Blender から Deadline Cloud にジョブを送信するには

1. Blender ファイルを保存します。
2. レンダリングメニューで、Deadline Cloud に送信を選択します。
 - GUI 依存関係をインストールするためのポップアップが表示される場合があります。OK を選択し、ダイアログが消えるのを待ってから、もう一度「Deadline Cloud に送信」を選択します。
3. ダイアログのタブを使用してジョブをカスタマイズします。
4. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
5. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

Blender 固有の設定

ジョブ固有の設定タブには、Blender で作成されたジョブに固有のオプションがあります。

Submit to AWS Deadline Cloud

Shared job settings **Job-specific settings** Job attachments Host requirements

Project Path

Output Directory ...

Output File Prefix

Scene ▾

Render Engine ▾

View Layers ▾

Cameras ▾

Cycles GPU Rendering ▾

Override Frame Range

✓ my-profile ▾

Settings... Export bundle Submit

- プロジェクトパス - 現在のプロジェクトが保存される場所。この値は変更できません。
- 出力ディレクトリ - レンダリングジョブからのファイル出力を保存する場所。
- 出力ファイルのプレフィックス - ファイル出力に名前を付けるときに使用するパターンは、ファイル名に関する Blender の規則に従います。出力ファイルは のようにフォーマットされます [LayerName]_[CameraName]_[OutputPrefix].[EXT]。
- シーン - レンダリングする現在のプロジェクトのシーン。
- レンダリングエンジン - 使用するレンダリングエンジン (サイクル、EVEE、またはワークベンチ)。
- レイヤーの表示 - レンダリングするレイヤー、またはシーン内の該当する各レイヤーを個別にレンダリングする「すべてのレンダリング可能なレイヤー」。
- カメラ - レンダリングするカメラ、シーン内の各カメラを個別にレンダリングする「すべてのレンダリング可能なカメラ」、またはタイムラインマーカーにバインドされたシーンのデフォルトのカメラを使用する「デフォルトのカメラを使用する」。
- サイクル GPU レンダリング - GPU レンダリングを有効にするかどうか。Blender でサポートされているデバイスタイプを選択するか、独自のデバイスタイプを指定します。このデバイスタイプがレンダリングマシンでサポートされていない場合、アダプターは CPU レンダリングにフォールバックする前に互換性のあるデバイスタイプを使用しようとしています。
- フレーム範囲の上書き - シーンファイルで設定されているものとは異なるフレームまたはフレーム範囲をレンダリングするには、このオプションを選択します。フレーム範囲は [Open Job Description](#) パターンに従います。

他の送信者タブの詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアのバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとしています。Blender のワークステーションバージョンが上記のバージョンテーブルに表示されない場合、これは失敗します。

サポートされていないバージョンの Blender が必要な場合は、次のオプションがあります。

- Blender からジョブを送信するときに、CondaPackages キューパラメータを上書きして、ワーカーで使用するサポートされているバージョンを指定できます (例: blender=4.5, blender-openjd=*). これは、シーンで使用される機能と、Blender がワークステーションバージョンのシーンとどのように連携するかに応じて、機能する場合と機能しない場合があります。
- ワーカーにインストールする目的のバージョンのカスタム conda レシピとチャンネルを構築できます。以下でリンクされているサポートされているバージョンの conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

Blender レンダーエンジン

Blender には、サポートされているいくつかの組み込みレンダーエンジンが含まれています。

レンダリングエンジン	説明	GPU サポート	注意事項
サイクル	物理ベースのパストレーサー	GPU/CPU ハイブリッド	GPU アクセラレーションによる本番稼働品質のレンダリング
Eevee	リアルタイムレンダリングエンジン	GPU 最適化	高速ビューポートと最終レンダリング
Workbench	ソリッドシェーディングエンジン	GPU 最適化	ワークフローのモデリングとスカルプティング用

すべてのレンダリングエンジンは、Blender 統合送信者によって自動的に検出され、設定されます。GPU アクセラレーションは、GPU 対応インスタンスでサービスマネージドフリートを使用する場合に使用できます。

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [Deadline Cloud for Blender](#)

- [Blender Conda レシピ](#)は、サポートされているバージョンの GitHub で利用できます。

エピック Unreal エンジン

Unreal Engine は、フォトリアルなビジュアルと没入型エクスペリエンスのためのリアルタイムの 3D 作成ツールです。Unreal Engine は、送信者、conda パッケージ、レンダリングパフォーマンスを向上させるアダプターを備えた Deadline Cloud でサポートされています。このガイドでは、Deadline Cloud と Unreal Engine を使用して、レンダリングタスクを複数のマシンに分散することで、映画レンダリングキュープロジェクトをより迅速にレンダリングするためのstep-by-stepの手順を示します。

サポートの概要

Unreal Engine は、次のコンポーネントでサポートされています。

- 送信者: Unreal Engine から直接ジョブを送信するための統合送信者プラグイン。シーンとアセットの自動検出機能を備えています。
- Conda パッケージ: サービスマネージドフリートへの自動インストールの Deadline Cloud。
- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows のみの送信者とワーカーのサポート。
- Movie Render Queue Integration: Unreal の Movie Render Queue システムのサポート。

Unreal Engine バージョンの互換性

次の表は、Unreal Engine バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート
5.4	Windows	Windows
5.5	Windows	Windows
5.6	Windows	Windows
5.7	Windows	Windows

Deadline Cloud Conda チャンネル

次の表に、conda チャンネルのサービスマネージドフリートで使用できる Unreal Engine に適用されるすべての deadline-cloud conda パッケージを示します。

OS	パッケージ	バージョン
Windows	unreal-engine	5.4
Windows	unreal-engine	5.5
Windows	unreal-engine	5.6
Windows	unreal-engine	5.7
Windows	unreal-engine-openjd	

開始方法

前提条件

Unreal Engine 送信者をインストールする前に、以下があることを確認してください。

- Windows ワークステーション (Windows 10 以降)
- サポートされているバージョンの Unreal Engine がインストールされている
- Deadline Cloud Monitor のインストール ([ここでダウンロード](#))
- GPU 対応の Windows サービスマネージドフリート、または Unreal Engine、Unreal Engine アダプター、ライセンス設定を備えたカスタマーマネージドフリートのいずれかを使用した Deadline Cloud ファームへのアクセス

送信者のインストール

Unreal Engine 送信者は Deadline Cloud 機能を Unreal Engine のプラグインとして追加し、映画レンダリングキュージョブを Deadline Cloud に直接送信してレンダリングできるようにします。

ブランチの選択

デプロイに適したブランチを選択します。

ブランチ	安定性	ユースケース	以下のために使用するをお勧めします
リリース	Stable	本番稼働	ほとんどのユーザー
メインライン	最新の機能	開発とテスト	上級ユーザー

i Tip

本番環境のリリースブランチを使用して、安定性を確保します。

Unreal をインストールする新しい Windows Amazon EC2 インスタンスの作成 (オプション)

新しい Windows Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを送信者として設定する場合、200 GB のストレージを持つ g5.2xlarge インスタンスは妥当な最小値です。

- 有効なインスタンスプロファイルを使用して Amazon EC2 インスタンスを起動します。これは、以下に示すように NVIDIA GRID ドライバーをダウンロードするために必要です。
- エピックインストーラをダウンロードし、サポートされているバージョンの Unreal (5.4 - 5.7) をインストールします。
 - DirectX 11 プラグインで実行する場合、UE 5.5 には既知のクラッシュバグがあります (「UE issue #UE-276282」を参照)。UE 5.5 で DirectX サポートが必要な場合は、DirectX 12 以降を使用します。
- NVIDIA GRID ドライバー - [Windows のインストール手順に従います](#)。

Windows のロングパス

以下の手順の多くは、デフォルトの Windows 最大パス長を超えるファイルを作成する場合があります。Deadline Cloud for Unreal Engine 送信者またはアダプターを Windows マシンにビルドしてインストールする前に、Windows のロングパスサポートを有効にすることをお勧めします。これを行うには、「[ファイルパスの最大数の制限](#)」ページの手順に従います。たとえば、[PowerShell コマンド](#)を実行します。

Windows のロングパスをサポートするように設定されていない依存関係が原因で [ワーカーエージェントに未解決の問題](#) もあります。ワーカーを設定するときは、リンクされた問題の回避策ステップに従って、問題が解決されるまで Windows のロングパスを完全にサポートします。

ビルドツールのインストール

現在、Unreal 送信者プラグインはローカルでコンパイルする必要があります。

1. Visual Studio [Installer を使用して Visual Studio](#) をインストールします。
2. [エピック互換性テーブルを確認して、Visual Studio とビルドツールのバージョンが Unreal のバージョンと互換性があることを確認](#) します。
3. 個々のコンポーネントで、選択した MSVC ビルドツールのバージョン (デフォルトでは「最新」) がテーブルの推奨バージョンと一致していることを確認します。互換性ガイダンスでは「またはそれ以降」のバージョンを提案している場合がありますが、「推奨」としてリストされているバージョンよりも新しいバージョンを使用する場合にビルドエラーが発生することがあります。
4. 個々のコンポーネントで、最新の .NET Framework SDK (4.6.1 と 4.8.1 が検証済み) を選択します。
5. ワークロードで、C++ を使用したデスクトップ開発を選択します。

Deadline Cloud Monitor のインストール

Deadline Cloud モニターは、Deadline Cloud にジョブを送信するための認証情報を管理し、ジョブのステータスをモニタリングするために使用されます。

1. [Deadline Cloud モニターのインストール手順に従](#)います。
2. サインインする。

環境設定

1. (まだインストールされていない場合) すべてのユーザーに最新バージョンの Python をインストールします (3.12 で検証済み)。
2. 環境変数が正しく設定されていることを確認します。システム環境変数では、PATH に以下を含める必要があります。
 - Python インストールへのパス (例: C:\Program Files\Python312)。
 - Python Scripts フォルダへのパス (例: C:\Program Files\Python312\scripts)。

- Unreal バイナリへのパス (例: C:\Program Files\Epic Games\UE_5.5\Engine\Binaries\Win64)。

Deadline Cloud ソフトウェアのインストール

最新のテスト済みリリースが必要か、最新のコミットのすべてが必要かに応じて、リリースブランチまたはメインライン `deadline-cloud-for-unreal-engine` からクローンまたはダウンロードします。

```
git clone https://github.com/aws-deadline/deadline-cloud-for-unreal-engine.git
cd deadline-cloud-for-unreal-engine
git switch release
```

オプション - スクリプトを使用してプラグインを構築およびインストールする

のヘルプスクリプトは `scripts/build_plugin.py`、オプションで次の 2 つのステップを自動化します。Unreal の最新バージョンの検索を試み、プラグインと Python の依存関係を構築し、正しい場所にインストールします。Unreal バージョンなどの設定を上書きできます。完全なヘルプリストを表示するには、以下を実行します。

```
python scripts/build_plugin.py -h
```

最新の Unreal Engine インストールを使用しての現在のコピーを送信者 `deadline-cloud-for-unreal-engine` としてビルドおよびインストールするには、以下を実行します。

```
python scripts/build_plugin.py --install
```

このスクリプトで を正常にインストールした場合は、「」に進むことができます [フリートの作成](#)。

プラグインの構築

Unreal のインストール場所と のインストール場所に基づいて、以下の最初の 2 つのパスを調整します `deadline-cloud-for-unreal-engine`。

Unreal Install Batchfiles フォルダから (`package` パラメータは任意の新しいディレクトリにすることができますが、UnrealDeadlineCloudService 後で呼び出す必要があります)。

```
cd C:\Program Files\Epic Games\UE_5.5\Engine\Build\BatchFiles
```

```
runuat.bat BuildPlugin -plugin="C:\deadline\deadline-cloud-for-unreal-engine\src\unreal_plugin\UnrealDeadlineCloudService.uplugin" -package="C:\UnrealDeadlineCloudService"
```

上記の「パッケージ」フォルダを Unreal インストールのプラグインフォルダ (例:) にコピーしますC:\Program Files\Epic Games\UE_5.5\Engine\Plugins\UnrealDeadlineCloudService。

Python の依存関係

必要な Python 依存関係をインストールする方法は 4 つあります。

1. 上記のリリースブランチからプラグインを構築してインストールした場合は、pip から をインストールできます。次の install コマンドを使用して、Unreal インストールへのパスを調整します。

```
"C:\Program Files\Epic Games\UE_5.5\Engine\Binaries\ThirdParty\Python3\Win64\python" -m pip install deadline-cloud-for-unreal-engine --target "C:\Program Files\Epic Games\UE_5.5\Engine\Plugins\UnrealDeadlineCloudService\Content\Python\libraries"
```

2. または、.uplugin ファイル (上記のステップでは に存在しますC:\Program Files\Epic Games\UE_5.5\Engine\Plugins\UnrealDeadlineCloudService\UnrealDeadlineCloudService.uplugin) に、GitHub/PyPI deadline-cloud-for-unreal-engineの の最新リリースに一致するPythonRequirementsセクションを追加できます。次に例を示します。

```
"PythonRequirements":  
[  
  {  
    "Platform": "All",  
    "Requirements":  
    [  
      "deadline-cloud-for-unreal-engine>=0.5.0"  
    ]  
  }  
]
```

Unreal の Python 設定で「厳密なハッシュ」機能を無効にしたり、使用する特定のライブラリと依存関係バージョンにハッシュ設定を追加したりできます。

3. メインラインからプルする場合、Python 依存関係がまだ PyPI にリリースされていない可能性があります。ローカルコピーを構築してインストールする必要があります。これは、ハッチで実行でき

ます。 .whl ファイルは、ハッチビルドによって出力されるバージョンを反映するように変更する必要があります。

```
pip install hatch
hatch build
"C:\Program Files\Epic Games\UE_5.5\Engine\Binaries\ThirdParty\Python3\Win64\python"
-m pip install dist\deadline_cloud_for_unreal_engine-0.2.2.post21-py3-none-any.whl --
target "C:\Program Files\Epic Games\UE_5.5\Engine\Plugins\UnrealDeadlineCloudService
\Content\Python\libraries"
```

4. 最後に、Python 依存関係は送信者インストーラによってインストールできます。これらはリリースブランチまたはメインラインブランチの上記のコードでは古くなっている可能性があるため、この方法は現時点では推奨されません。

1. Deadline Cloud AWS コンソールのダウンロードタブから、またはワークステーションのセットアップ → ダウンロードの Deadline Cloud モニター内から、送信者インストーラをダウンロードします。
2. すべてのユーザーに対してインストーラを実行します。デフォルトのインストール場所は問題ありません。
3. Unreal Engine プラグインを有効にします。
4. Unreal Engine プラグインのインストールパスが、プラグインがコピーされた場所と一致していることを確認します (特に Unreal バージョンが一致していることを確認します)。

通知の更新

送信者プラグインは、Unreal Editor の起動時に GitHub で新しいリリースを自動的に確認します。更新が利用可能な場合、リリースページにアクセスするように求めるダイアログが表示されます。

更新通知を無効にするには、「Deadline Cloud settings」パネルの「General Settings」(Edit > Project Settings > Plugins > Deadline Cloud) で「Show submitter update notifications」のチェックボックスをオフにします。

または、CLI を使用することもできます。

```
deadline config set settings.submitter_update_notification false
```

を再度有効にするには:

```
deadline config set settings.submitter_update_notification true
```

フリートの作成

すでに Windows フリートがあり、新しいフリートを設定する必要がない場合は、[にスキップできます](#) [テストレンダーの送信](#)。

サービスマネージドフリート (SMF) の作成

まだ SMF がない場合は、[サービスマネージドフリートユーザーガイド](#)に従って SMF を作成します。[サービスマネージドフリート \(SMFs\)](#) では、Unreal Engine とアダプターは、[デフォルトのキュー環境](#)の `deadline-cloud conda` チャンネルを使用して自動的に使用できます。レンダリングを開始する準備が整いました。テストレンダージョブを送信するには、[テストレンダーの送信](#)以下に進みます。

カスターマネージドフリート (CMF) の作成

1. まだ CMF がない場合は、[「カスターマネージドフリートを作成する」](#)を参照してください。

Warning

CMF をキューに関連付ける場合は、デフォルトの `conda` キュー環境を削除します。これにより、`conda` 環境が使用されなくなり、CMF に送信されたジョブにデフォルトの SMF 固有の変数が誤って使用されます。CMF で `conda` を使用する場合は、ジョブの送信中にパラメータ定義オーバーライドの変数 `CondaPackages` と `CondaChannels` 変数を必ず更新してください。

2. [ワーカーホストのセットアップと設定](#)に従って、ワーカーホストをセットアップします。
3. [「Windows ジョブユーザーシークレットへのアクセスの管理」](#)に従って、CMF ワーカーの Windows ジョブユーザーシークレットを設定します。
4. [ジョブに必要なソフトウェアをインストールして設定](#)し、ジョブの実行に必要なソフトウェアをインストールします。
5. に従って[カスターマネージドフリート \(CMF\) ワーカーのセットアップ](#)、Unreal Engine ジョブを実行するようにワーカーノードを設定します。

テストレンダーの送信

この例では、Unreal Marketplace の Meerkat デモを使用しています。

1. エピックゲームランチャーを起動します。
2. サンプルタブから Meerkat デモをインストールします。
3. Meerkat デモからプロジェクトを作成し、開きます。
4. 編集メニューからプラグインを選択し、UnrealDeadlineCloudService を検索して有効にします。
5. プラグインを初めて有効にした場合は、Unreal を再起動します。
6. Edit > Project Settings で、Movie Render Pipeline セクションを検索します。
 - デフォルトのリモートエグゼキュターで、MoviePipelineDeadlineCloudRemoteExecutor を選択します。
 - デフォルトのエグゼキュタージョブで、MoviePipelineDeadlineCloudExecutorJob を選択します。
 - デフォルトのジョブ設定クラスで、追加アイコンを選択し、DeadlineCloudRenderStepSetting を追加します。
7. Deadline Cloud 設定を検索し、認証を確認します。
 - ステータスが「AUTHENTICATED」と表示され、Deadline Cloud API が「AUTHORIZED」と表示されていることを確認します。
 - 表示されない場合は、まずログインボタンを使用してみます。これが機能しない場合は、Deadline Cloud Monitor を開き、ログインしていることを確認します。
 - 「Deadline Cloud Workstation Configuration」セクションで、次の操作を行います。
 - グローバル設定で、AWS プロファイルが Deadline Cloud モニタープロファイルに正しく設定されていることを確認します。
 - プロファイルで、デフォルトのファームがファームに設定されていることを確認します。
 - Farm で、デフォルトキューが、上記で設定したフリートに関連付けられているキューに設定されていることを確認します。
8. プロジェクト設定ウィンドウを終了します。
9. Windows、Cinematics、Movie Render Queue を選択します。
 - + Render を選択し、Main_SEQ を選択します。
 - 設定列で UnsavedConfig を選択します。
 - ポップアップウィンドウの左側に Deadline Cloud の設定が表示されます。その後、このウィンドウを閉じることができます。

- ダイアログの右側で、ジョブ設定を行います。
- プリセットオーバーライド (このダイアログを広げる必要がある場合があります) の下:
 - ジョブ共有設定を展開します。
 - [Name] を [Unreal Test Job] に設定します。
 - 最大再試行回数を 2 に設定します。
 - ジョブアタッチメントを展開します。
 - 入力ファイルで、自動検出の表示を選択します。
 - 自動検出されたファイルのリストが正しく入力されていることを確認します。
 - ジョブテンプレートの上書き:
 - 5.6 とは異なるバージョンを使用している場合は、CondaPackages で Unreal Engine バージョンを更新します。
 - Unreal Engine バージョンの自動検出は、今後のリリースで予定されています。
- レンダリング (リモート) を選択します。

10Deadline Cloud モニターに移動し、ジョブの進行状況を確認できます。

カスタマーマネージドフリート (CMF) ワーカーのセットアップ

このセクションでは、Unreal Engine を使用した Deadline Cloud の CMF ワーカーとして Amazon EC2 インスタンスを設定する手順について説明します。

概要

CMF と SMF の主な違いは次のとおりです。

- CMF: Unreal Engine とアダプターをワーカーホストに手動でインストールします。
- SMF: deadline-cloud conda チャンネルを介した自動可用性。

ブランチの選択

デプロイに適したブランチを選択します。

ブランチ	安定性	ユースケース
リリース	Stable	本番稼働用デプロイ

ブランチ	安定性	ユースケース
メインライン	最新	開発とテスト

Important

互換性の問題を避けるため、ワーカーのバージョンが送信者のバージョンと一致していることを確認してください。

Amazon EC2 インスタンスのセットアップ

推奨されるインスタンス設定は次のとおりです。

- インスタンスタイプ: g5.2xlarge 以上。
- ストレージ: 最小 200 GB。
- OS: Windows Server 2019/2022。

ソフトウェアのインストール

1. Unreal Engine のインストール:

1. エピックゲームランチャーをダウンロードします。
2. Unreal Engine 5.4 以降をインストールします。

Note

Deadline Cloud の互換性には Unreal Engine 5.4 以降が必要です。

2. NVIDIA GRID ドライバーをインストールします。

- [AWS NVIDIA GRID ドライバーのインストールガイド](#)に従ってください。
- Amazon EC2 インスタンスでの GPU アクセラレーションレンダリングに必要です。

ビルドツールのインストール

現在、Unreal プラグインはローカルでコンパイルする必要があります。

1. Visual Studio [Installer](#) を使用して [Visual Studio](#) をインストールします。
2. [エピック互換性テーブルを確認して、Visual Studio とビルドツールのバージョンが Unreal のバージョンと互換性がある](#)ことを確認します。
3. 個々のコンポーネントで、選択した MSVC ビルドツールのバージョン (デフォルトでは「最新」) がテーブルの推奨バージョンと一致していることを確認します。
4. 個々のコンポーネントで、最新の .NET Framework SDK を選択します (4.6.1 と 4.8.1 は検証済みです)。
5. ワークロードで、C++ を使用したデスクトップ開発を選択します。

Deadline Cloud ソフトウェアのインストール

リリースブランチまたはメインライン `deadline-cloud-for-unreal-engine` からクローンを作成するか、ダウンロードします。ワーカーバージョンのライブラリが、送信者が使用しているバージョンと互換性があることを確認します。

```
git clone https://github.com/aws-deadline/deadline-cloud-for-unreal-engine.git
cd deadline-cloud-for-unreal-engine
git switch release
```

オプション - スクリプトを使用してプラグインと依存関係を構築およびインストールします。

残りのインストールステップを自動化 `scripts/build_plugin.py` できるヘルパースクリプトがあります。Unreal の最新バージョンを見つけ、プラグインと Python の依存関係を構築し、正しい場所にインストールしようとしています。使用する Unreal バージョンなどの設定は上書きできます。完全なヘルプリストを表示するには、以下を実行します。

```
python scripts/build_plugin.py -h
```

最新の Unreal Engine インストールを使用して、ワーカー `deadline-cloud-for-unreal-engine` としての現在のコピーを構築してインストールするには、以下を実行します。

```
python scripts/build_plugin.py --install --worker
```

以下を実行して Deadline Cloud ワーカーエージェントを設定します。

```
install-deadline-worker ^
--farm-id FARM_ID ^
--fleet-id FLEET_ID ^
--region REGION ^
--allow-shutdown
```

このスクリプトで をインストールし、ワーカーエージェントを正常に設定した場合は、「」に進むことができます [Deadline Cloud ワーカーエージェントサービスの開始](#)。

```
python -m pip install deadline-cloud-worker-agent
```

使用する送信者のバージョンに応じて、正しいバージョンのアダプターをインストールする必要があります。GitHub のリリースブランチから送信者のバージョンを使用している場合は、pip を使用してインストールできます。

```
python -m pip install deadline-cloud-for-unreal-engine
```

送信者のメインラインまたはカスタム開発中バージョンを使用している場合は、互換性の問題を避けるため、同じバージョンのコードからビルドしてインストールするか、送信者ビルドから .whl ファイルを転送することをお勧めします。

```
pip install hatch
hatch build
python -m pip install dist\my-built-wheel.whl
```

プラグインの構築

Unreal のインストール場所と のインストール場所に基づいて、以下の最初の 2 つのパスを調整します deadline-cloud-for-unreal-engine。

Unreal Install Batchfiles フォルダから (packageパラメータは任意の新しいディレクトリにすることができますが、UnrealDeadlineCloudService後で呼び出す必要があります)。

```
cd C:\Program Files\Epic Games\UE_5.5\Engine\Build\BatchFiles
runuat.bat BuildPlugin -plugin="C:\deadline\deadline-cloud-for-unreal-
engine\src\unreal_plugin\UnrealDeadlineCloudService.uplugin" -package="C:
\UnrealDeadlineCloudService"
```

上記の「パッケージ」フォルダを Unreal インストールのプラグインフォルダ (例:) にコピーしますC:\Program Files\Epic Games\UE_5.5\Engine\Plugins\UnrealDeadlineCloudService。

pywin32

Unreal のバージョンの Python には pywin32 が必要です。Unreal のサードパーティー Python インストールのコピーを使用して Pip インストールします。

```
"C:\Program Files\Epic Games\UE_5.5\Engine\Binaries\ThirdParty\Python3\Win64\python" -m pip install pywin32
```

Deadline Cloud ワーカーエージェントサービスの開始

CMF ワーカーインスタンスの場合:

1. Task Manager を開きます。
2. 右側のサービスタブを選択します。
3. DeadlineWorker を検索します。
 - リストにない場合は、[CMF ホストのセットアップ](#)ステップのステップ (install-deadline-worker 特に) を見落としている可能性があります。
4. サービスのステータスが現在「実行中」でない場合は、右クリックして開始を選択します。
5. DeadlineWorker サービスが起動しない場合は、以下の場所でワーカーエージェントの起動ログを確認してください。
 - C:\ProgramData\Amazon\Deadline\Logs\worker-agent.log
 - C:\ProgramData\Amazon\Deadline\Logs\queue-<queueid>\session-<sessionid>.log

認証情報管理を実施する

このセクションでは、Perforce を Deadline Cloud ワーカーと統合するための安全な認証情報管理について説明します。

概要

Deadline Cloud ワーカーの Perforce 認証情報を設定する方法はいくつかあります。各方法には、さまざまなセキュリティ上の影響とユースケースがあります。

方法	セキュリティレベル	デプロイのサポート	推奨用途
AWS Secrets Manager (Secrets Manager)	高	SMF + CMF	本番稼働用 (推奨)
ジョブの環境変数	低	SMF + CMF	開発とテストのみ
キュー環境変数	低	SMF + CMF	開発とテストのみ
Windows レジストリ	中	CMF のみ	レガシー CMF 設定
事前設定された管理者ユーザー	中	CMF のみ	CMF 設定の簡素化

Important

本番環境には Secrets Manager を使用します。一元化された暗号化された認証情報ストレージと監査証跡を提供し、SMF と CMF の両方のデプロイで動作します。

P4 認証情報の基本

Perforce サーバーの URL とポート、ユーザー名、パスワードなどの接続設定を取得するには、Perforce は次の優先度に従います。

1. Perforce (P4) の任意のフレームワークの接続パラメータ (p4python.P4 など)。
2. ユーザー/システム環境変数: P4PORT、P4USER、P4CLIENT、P4PASSWD。
3. Windows レジストリ: HKEY_LOCAL_MACHINE\SOFTWARE\Perforce\Environment (システム全体の設定) または HKEY_CURRENT_USER\SOFTWARE\Perforce\Environment (ユーザー固有の設定)。

これらの優先順位で、次の設定がある場合:

- 接続パラメータpasswordは `password.from.connection`
- 環境変数は `%P4PORT%` です。 `ssl:perforce.from.env:1666`
- Windows レジストリ `HKEY_CURRENT_USER\SOFTWARE\Perforce\Environment:P4PORT` は `ssl:perforce.from.registry:1666`
- Windows レジストリ `HKEY_CURRENT_USER\SOFTWARE\Perforce\Environment:P4USER` は `user.from.registry`
- Windows レジストリ `HKEY_CURRENT_USER\SOFTWARE\Perforce\Environment:P4PASSWD` は `password.from.registry`

結果の Perforce 接続は次のとおりです。

```
Port = ssl:perforce.from.env:1666
User = user.from.registry
Password = password.from.connection
```

Secrets Manager (推奨)

Secrets Manager は、CMF と SMF の両方のデプロイに対して最も安全でスケーラブルなソリューションを提供します。

このアプローチには以下の利点があります。

- 一元化されたセキュリティ: すべての P4 認証情報を 1 つの暗号化された場所に保存します。
- 認証情報が公開されない: 認証情報がジョブ設定やログに表示されることはありません。
- 自動ローテーション: ジョブの再設定なしで認証情報ローテーションをサポートします。
- 監査証跡: 認証情報のアクセスと使用状況を追跡します。
- ユニバーサルサポート: CMF と SMF の両方のデプロイで動作します。
- ログの秘匿化: ジョブログで自動的に秘匿化された接続認証情報。

ステップ 1: Secrets Manager でシークレットを作成する

Perforce 接続パラメータを含むシークレットを作成します。

必要なキーと値のペア:

- P4PORT - サーバー URL とポートを実行します。
- P4USER - ユーザー名を強制します。

- P4PASSWD - パスワードを強制します。

Important

キー名は、 を操作するために P4 接続パラメータと完全に一致する必要があります `deadline-cloud-for-unreal-engine`。

シークレットの例:

```
{
  "P4PORT": "ssl:your-perforce-server.com:1666",
  "P4USER": "your-perforce-username",
  "P4PASSWD": "your-perforce-password"
}
```

シークレットを作成するには:

1. Secrets Manager コンソールを開きます。
2. 「新しいシークレットを保存する」を選択します。
3. 「その他のタイプのシークレット」を選択します。
4. 上記のキーと値のペアを入力します。
5. シークレットに名前を付けます (例: `deadline-cloud-p4-credentials`)。
6. 作成プロセスを完了します。

ステップ 2: ワーカーにシークレットへのアクセスを許可する

ワーカーはシークレットにアクセスするためのアクセス `secretsmanager:GetSecretValue` 許可が必要です。これは、[Windows ジョブのユーザーシークレットの管理](#)と同じパターンに従います。

アクセスを許可するには:

1. Secrets Manager コンソールを開き、シークレットに移動します。
2. 「リソースのアクセス許可」セクションで、次のポリシーを追加します。

```
{
  "Statement": [
```

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "QUEUE_ROLE_ARN"
  },
  "Action": [
    "secretsmanager:GetSecretValue"
  ],
  "Resource": "*"
}
```

3. を実際のキューロール ARN `QUEUE_ROLE_ARN`に置き換えます。
4. ポリシーを保存します。

Note

シークレット名を保存します。P4 レンダリングジョブを設定するときに必要になります。次のステップ [Perforce レンダリングジョブの作成](#)については、「」を参照してください。

ジョブの環境変数

Warning

このアプローチは、ジョブ設定とログで認証情報を公開するため、本番環境にはお勧めしません。本番環境には Secrets Manager を使用します。

接続認証情報は、[p4_sync_smf_environment](#)、[ugs_sync_smf_environment](#)、または CMF の同様の環境など、ワークスペースの作成が行われるジョブ環境内で渡すことができます。または、新しい環境テンプレートを作成し、ジョブに付加します。

```
name: P4Credentials
variables:
  P4PORT: ssl:my-perforce.com:1666
  P4USER: j.doe
  P4PASSWD: MyVeRyS3cretP4ssW0rd
```

このアプローチには、次のセキュリティリスクがあります。

- 認証情報はジョブ設定に表示されます。
- パスワードはログに表示される場合があります (自動的に編集されません)。
- 一元化された認証情報管理はありません。
- 認証情報のローテーションが困難。

キュー環境変数

⚠ Warning

このアプローチは、認証情報をキュー設定に保存するため、本番環境にはお勧めしません。Secrets Manager を使用して、安全な認証情報を管理します。

[Deadline Cloud ユーザーガイド](#)によると、キュー環境を使用して、ソフトウェアアプリケーション、環境変数、その他のリソースをキュー内のジョブに提供できます。キュー環境のサンプルは、[deadline-cloud-samples の queue_environments フォルダ](#)にあります。

Deadline Cloud モニターまたはコンソールを使用してキュー環境を追加する

1. Deadline Cloud モニターまたは AWS コンソールを開きます。
2. 作業しているファームとキューに移動します。
3. キュー環境タブを選択します。
4. Action を選択し、Create new with YAML を選択します。
5. 以下を追加して保存します。

```
specificationVersion: environment-2023-09
name: P4Credentials
variables:
  P4PORT: ssl:my-perforce.com:1666
  P4USER: j.doe
  P4PASSWD: MyVeRyS3cretP4ssW0rd
```

CLI を使用してキュー環境を追加する

1. 上記のサンプルを使用して p4_credentials.yaml ファイルを作成します。

2. 次の CLI コマンドを実行します。

```
aws deadline create-queue-environment \  
  --farm-id FARM_ID \  
  --queue-id QUEUE_ID \  
  --priority 1 \  
  --template-type YAML \  
  --template file://p4_credentials.yaml
```

Windows レジストリ (CMF のみ)

Important

このソリューションは、ワーカーホストを直接設定できる CMF にのみ適しています。SMF デプロイの場合は、Secrets Manager を使用します。

Windows レジストリは、ワーカーマシンにローカル認証情報ストレージを提供します。このメソッドは、認証情報が保存されている標準の Perforce 優先度システムを使用します。

- HKEY_LOCAL_MACHINE\SOFTWARE\Perforce\Environment (システム全体の設定)。
- HKEY_CURRENT_USER\SOFTWARE\Perforce\Environment (ユーザー固有の設定)。

このアプローチは、ワーカーマシンの設定に直接アクセスできる CMF デプロイにのみ適しています。

事前設定された管理者ユーザー (CMF のみ)

Important

このソリューションは、ワーカーホストを直接設定できる CMF にのみ適しています。SMF デプロイの場合は、Secrets Manager を使用します。

有効期限が切れていない接続を持つレンダーノード専用の P4 ユーザーを作成し、ユーザー名やパスワードなどのシークレットデータを渡す必要がなくなります。この場合、1 人の専用 P4 ユーザーを使用して、プロジェクトデポにアクセスできるコミット (マスター) サーバーやエッジサーバーなど、すべての P4 サーバーに接続できます。

したがって、デフォルトがワーカーで設定されていない場合は、接続先のポートを渡すだけで済みます。これは、上記の手順と同様に、ジョブ環境またはキュー環境に以下を追加することで実現できます。

```
name: P4Sync
variables:
  P4PORT: ssl:my-perforce.com:1666
```

Perforce レンダリングジョブの作成

このセクションでは、Perforce 統合レンダージョブデータアセットを設定して、Perforce 統合レンダージョブを Unreal Engine から Deadline Cloud に送信できるようにする方法について説明します。

前提条件

Perforce リポジトリで Unreal Engine から映画レンダリングキュー (MRQ) ジョブを送信する前に:

Unreal プロジェクトのセットアップ:

- プロジェクトは Perforce ワークスペース内にある必要があります。
- Perforce 接続が確立され、ログインしていることを確認します。

Deadline Cloud のセットアップ:

- [送信者のインストール](#) を完了します。
- ワーカー [認証情報管理を実施する](#) 用に を設定します。

Perforce の要件:

- プロジェクトファイルを含む有効な Perforce ワークスペース。
- 同期オペレーションの適切な Perforce アクセス許可。

レンダリングジョブアーキテクチャを実行する

P4 レンダリングジョブは、Perforce 同期機能を使用して標準レンダリングジョブを拡張します。

```
Deadline Cloud Perforce render job
```

```
### Environments
#   ### Apply Perforce Credentials from Secrets Manager
#   ### Sync Perforce Environment (CMF/SMF)
#   #   ### Initializes Perforce workspace and syncs repository on workers
#   ### Launch UE Environment
#       ### Starts Unreal Engine with Perforce workspace paths
### Steps
    ### Render Step
        ### Executes Movie Render Queue process
```

標準ジョブとの主な違い:

- 認証情報管理: Secrets Manager からの Secure Perforce 認証情報アプリケーション。
- リポジトリ同期: ワークスペース管理用の追加の Perforce 同期環境。
- パス解決: 環境変数は Perforce ワークスペースパスを参照します。
- 依存関係の収集: Perforce で追跡されたアセットの自動収集と同期。

Perforce レンダリングジョブコンポーネントのセットアップ

Perforce 統合レンダージョブに必要なデータアセットを作成するには、次の手順に従います。

1. Apply Perforce 認証情報データアセットを作成する

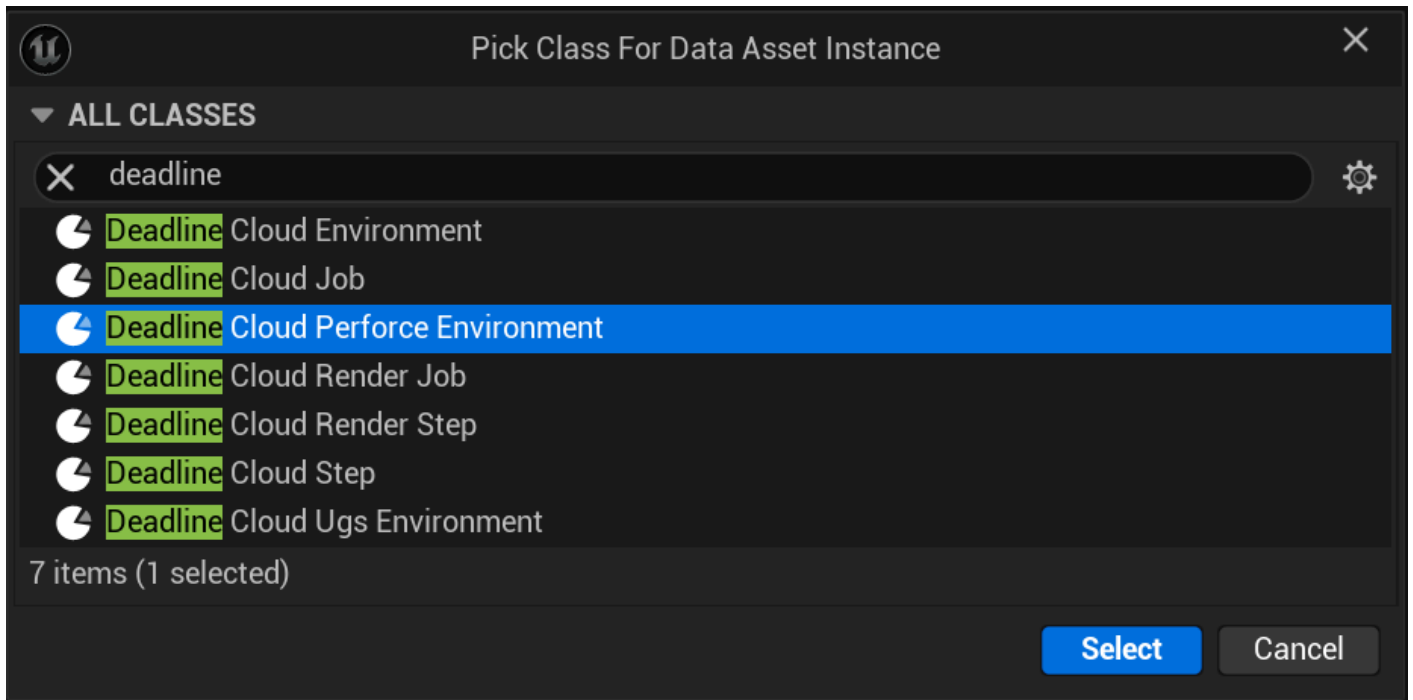
Secrets Manager から Perforce 認証情報を取得し、Perforce 接続に適用するための OpenJD 環境を設定します。

Note

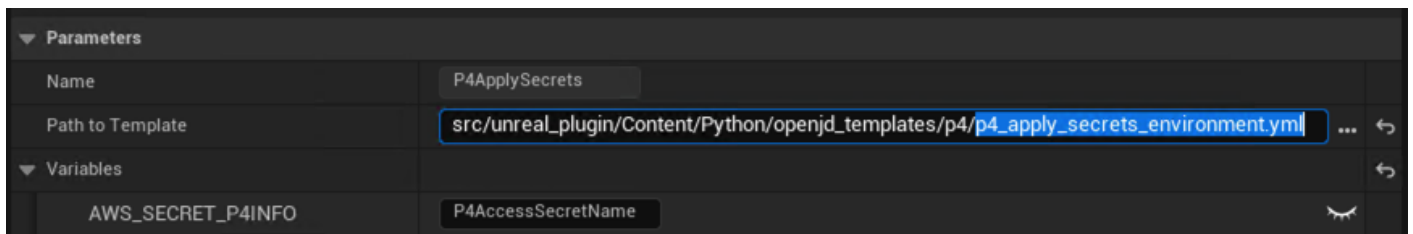
環境変数は `openjd_env` プレフィックスが付いたログに表示されますが、機密データ (ポート、ユーザー、パスワード) はセキュリティのために自動的に編集されます。

設定手順:

1. 新しい Deadline Cloud Perforce Environment データアセットを作成します。



2. データアセットにわかりやすい名前を付けます (例: ApplyP4SecretEnv)。
3. から `p4_apply_secrets_environment.yml` テンプレートを選択します `Content/Python/openjd_templates/p4/`。
4. シークレットリファレンスを設定します。
 - `AWS_SECRET_P4INFO` に Perforce 認証情報シークレット名を入力します。
 - これは、 で作成されたシークレット名と一致する必要があります [認証情報管理を実施する](#)。



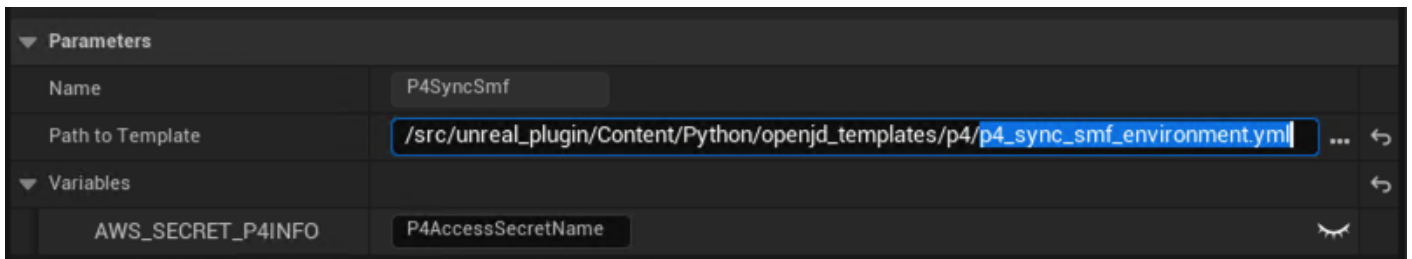
2. Perforce 同期環境データアセットを作成する

Perforce ワークスペースを作成し、Perforce サーバーからファイルを同期し、レンダリング後にワークスペースをクリーンアップするための OpenJD 環境を設定します。

設定手順:

1. 新しい Deadline Cloud Perforce Environment データアセットを作成します。
2. フリートタイプに基づいてデータアセットに名前を付けます。

- サービスマネージドフリート (SMF) の P4SyncSMFEnv。
 - カスタマーマネージドフリート (CMF) の P4SyncCMFEnv。
3. から適切なテンプレートを選択しますContent/Python/openjd_templates/p4/。
 - SMF: p4_sync_smf_environment.yml。
 - CMF: p4_sync_cmf_environment.yml。
 4. シークレットリファレンスを設定します。
 - AWS_SECRET_P4INFO に Perforce 認証情報シークレット名を入力します。
 - ステップ 1 のシークレットと一致する必要があります。

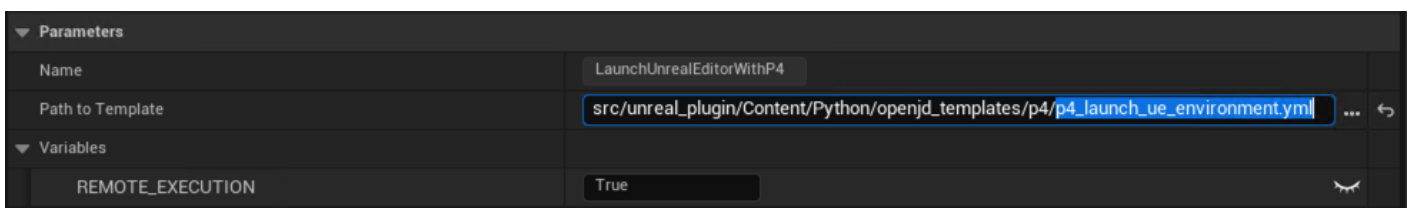


3. Perforce Launch UE 環境データアセットを作成する

Perforce 統合で Unreal Engine を起動するための OpenJD 環境を設定します。これにより、同期環境によって作成された Perforce ワークスペースが自動的に参照されます。

設定手順:

1. 新しい Deadline Cloud Environment データアセットを作成します。
2. データアセットにわかりやすい名前を付けます (例:P4LaunchUEEnv) 。
3. からp4_launch_ue_environment.ymlテンプレートを選択しますContent/Python/openjd_templates/p4/。
4. 環境設定を構成する:
 - REMOTE_EXECUTION を に設定しますTrue。
 - この設定により、リモートレンダリング機能が有効になります。

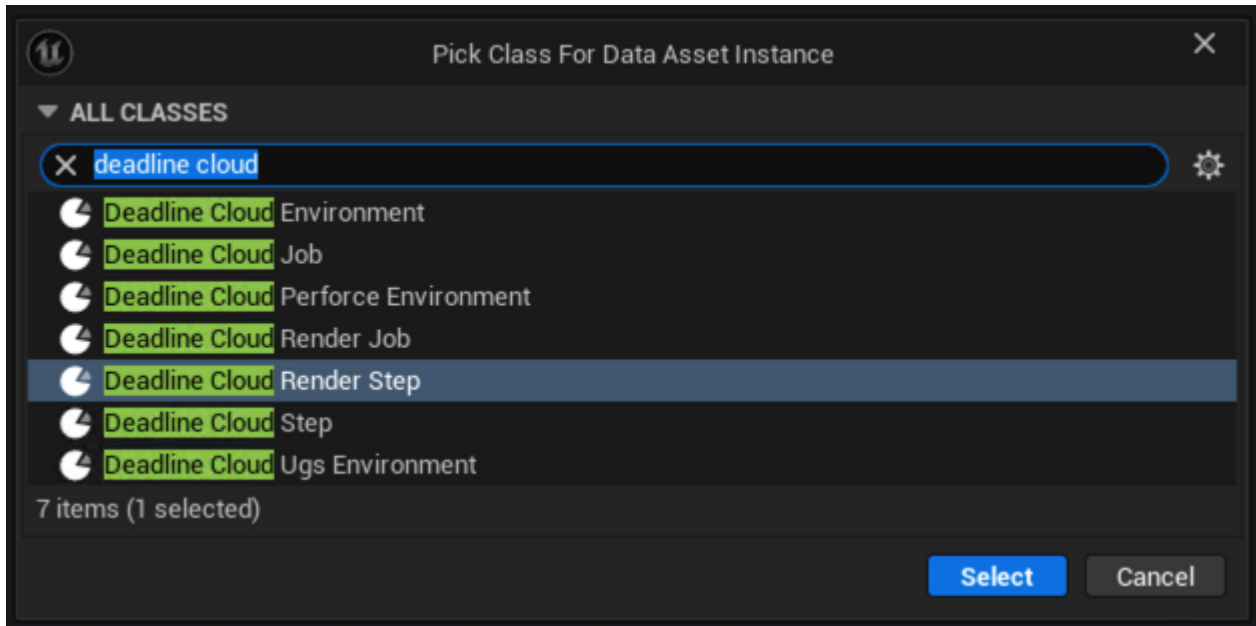


4. Perforce レンダリングステップデータアセットを作成する

レンダリングプロセスを実行するための OpenJD レンダリングステップを設定します。

設定手順:

1. 新しい Deadline Cloud Render Step データアセットを作成します。



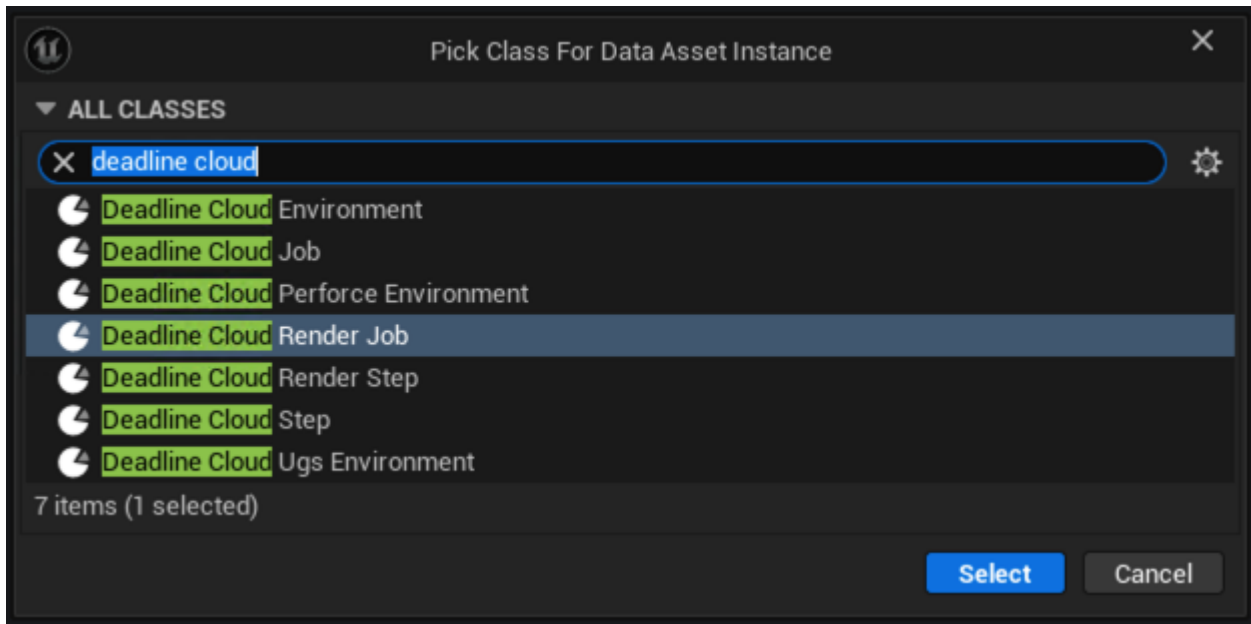
2. データアセットにわかりやすい名前を付けます (例: P4RenderStep)。
3. から p4_render_step.yml テンプレートを選択します Content/Python/openjd_templates/p4/。

5. Perforce レンダリングジョブデータアセットを作成する

レンダリングワークフロー全体をオーケストレーションする OpenJD レンダリングジョブを設定します。

設定手順:

1. 新しい Deadline Cloud Render Job データアセットを作成します。



2. データアセットにわかりやすい名前を付けます (例:「P4RenderJob」)。
3. から `p4_render_job.yml` テンプレートを選択します `Content/Python/openjd_templates/p4/`。
4. パラメータ定義の確認: テンプレートには、これらのパラメータとデフォルトの動作が含まれます。

パラメータ	説明	自動入力	必要なアクション
ProjectRelativePath	Perforce ワークスペースルートに対するプロジェクトパス	はい	空のままにする - 自動入力
ProjectName	Perforce ワークスペース作成のプロジェクト名	はい	空のままにする - 自動入力
PerforceChangelistNumber	ワークスペースを に同期するように変更リストを強制する	はい	空のままにする - 自動入力
PerforceWorkspaceSpecification	{workspace_name} トークンを使用してクライアント仕様を実行する	はい	空のままにする - 自動入力

パラメータ	説明	自動入力	必要なアクション
ionTemplate			
MrqJobDependenciesDescriptor	同期用の MRQ 依存関係を持つ JSON ファイル	はい	空のままにする - 自動入力
ExtraCmdArgsFile	追加引数のファイル (1024 文字の制限を回避)	いいえ	オプション - 標準セットアップにデフォルトを使用する
FramesPerTask	タスクごとにレンダリングするフレーム数	いいえ	オプション - デフォルト (0) を使用してタスクをショットで分割する
ExtraCmdArgs	追加の Unreal 起動引数	いいえ	オプション - 標準セットアップにデフォルトを使用する
Executable	レンダーノードの非現実的な実行可能ファイル名	いいえ	設定 - 標準セットアップにはデフォルトを使用します
CondaPackages	ジョブのレンダリングに必要な Conda パッケージ	いいえ	設定 - 標準セットアップにはデフォルトを使用します

パラメータ	説明	自動入力	必要なアクション
CondaChannels	パッケージが保存されている Conda チャンネル	いいえ	設定 - 標準セットアップにはデフォルトを使用します
ChunkSize	1 回のレンダリングセッションでグループ化されたショットの数	いいえ	設定 - デフォルト: 1 (パフォーマンスに合わせて調整)
MarketplacePluginsDir	エンジン Marketplace プラグインへのパス	はい	空のままにする - 自動入力

パラメータ設定ガイドライン:

- 自動入力パラメータ: これらは空のままにします。ジョブの送信中に自動的に入力されます。
- 手動パラメータ: デフォルトを確認し、特定の要件に基づいて調整します。
- ChunkSize: 1 から始めて、シンプルなショットでパフォーマンスを向上させます。

Parameter Name	Value
ProjectRelativePath	
ProjectName	
PerforceChangelistNumber	
PerforceWorkspaceSpecificationTemplate	
MrqJobDependenciesDescriptor	
ExtraCmdArgs	-log
ExtraCmdArgsFile	
Executable	UnrealEditor-Cmd
CondaPackages	unrealengine=5.6 unrealengine-openjd=0.6.*
CondaChannels	deadline-cloud
ChunkSize	1

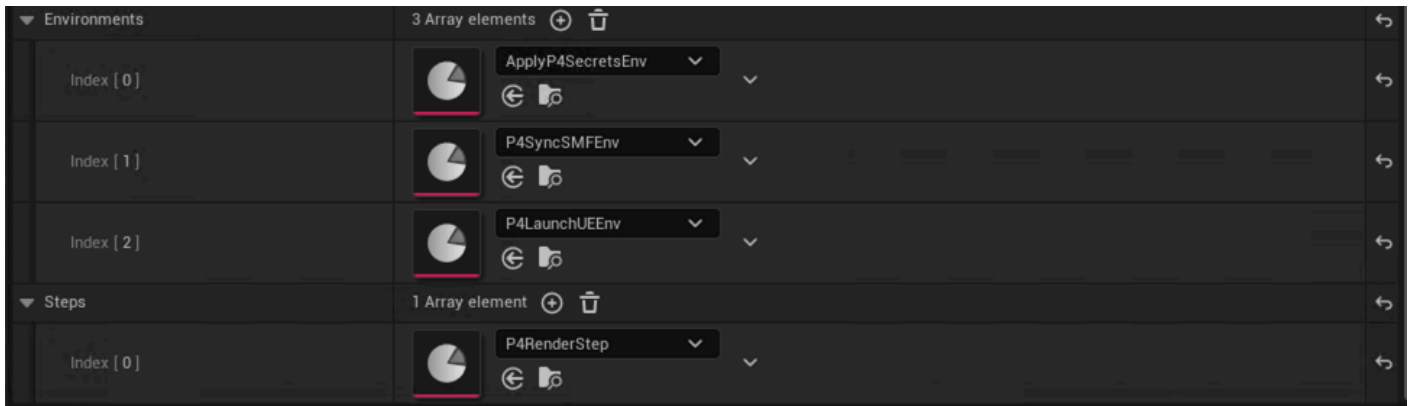
5. 環境を設定する (この正確な順序で):

Order	環境	目的
1 日	「ApplyP4SecretEnv」	Secrets Manager から Perforce 認証情報を適用する
2 番目	「P4SyncSMFEnv」 または「P4SyncCMFEnv」	Perforce ワークスペースとファイルを同期する
3 番目	「P4LaunchUEEnv」	Perforce パスを使用して Unreal Engine を起動する

⚠ Important

環境の順序は、適切な依存関係解決と認証情報フローに不可欠です。

6. レンダリングステップを追加する: ステップセクションに「P4RenderStep」を追加します。



ベストプラクティス

送信前チェックリスト

- 認証情報: 認証情報を適切に設定します (「」を参照[認証情報管理を実施する](#))。
- Workspace: 仕様には、必要なすべてのビューマッピングが含まれます。
- テスト: 大規模なジョブを送信する前に、小さなレンダリングタスクでテストします。

- 依存関係: すべてのプロジェクトの依存関係が Perforce にあり、アクセス可能であることを確認します。
- アクセス許可: ワーカーロールが Secrets Manager にアクセスできることを確認します。

パフォーマンスの最適化

チャンクサイズ設定:

チャンクサイズ	ユースケース	パフォーマンスへの影響
1~2 ショット	複雑なショット、詳細なレビューが必要	スループットの低下、品質管理の向上
4~8 ショット	バランスの取れたワークロード、一般的なプロジェクト	速度と管理可能性の最適なバランス
10 回以上	シンプルなショット、バッチ処理	最大スループット、最小オーバーヘッド

以下の追加の最適化を検討してください。

- 依存関係: 不要なアセット依存関係を最小限に抑えて、同期時間を短縮します。
- Workspace ビュー: Perforce ワークスペースビューを最適化して、必要なファイルのみを同期します。

モニタリングとデバッグ

- ジョブログ: ジョブ実行ログで Perforce 同期プロセスをモニタリングします。
- ワークスペースのステータス: ワークスペースの作成と同期の完了を確認します。
- 依存関係の収集: 必要なすべてのアセットがキャプチャされたことを確認します。
- パフォーマンスメトリクス: 同期時間とレンダリングパフォーマンスを追跡します。

トラブルシューティング

一般的な の問題と解決策

問題	症状	ソリューション
P4 接続の失敗	認証エラー、タイムアウト	Perforce 認証情報の検証、ネットワーク接続の確認、Perforce サーバーのアクセシビリティの検証
Workspace 同期エラー	同期失敗、アクセス許可拒否	Perforce ユーザーのアクセス許可をチェックする、ワークスペースの仕様を検証する、変更リストが存在することを確認する
依存関係がない	レンダリングの失敗、アセットの欠落	の確認 <code>MrqJobDependencies Descriptor</code> 、ソフトリファレンスの収集の確認、Perforce のすべてのアセットの検証
パス解決の問題	ファイルが見つかりませんエラー	<code>P4_CLIENT_DIRECTORY</code> 変数の検証、環境の順序の確認、ワークスペースのルートパスの検証

デバッグステップ

1. ジョブログを確認する: Perforce 同期環境ログで詳細なエラーメッセージを確認します。
2. ワークスペースの検証: Perforce ワークスペースの仕様が正しいことを確認します。
3. ローカルでテストする: 送信者マシンから Perforce オペレーションが機能することを確認します。
4. アクセス許可の確認: ワーカーロールにシークレットアクセスがあることを確認します (Secrets Manager を使用している場合)。

カスタムホストの要件

このセクションでは、Unreal Engine で Deadline Cloud レンダリングのカスタムホスト要件を作成して使用する方法について説明します。

概要

ホスト要件は、特定のレンダリングステップを実行する資格があるフリートを定義します。UDeadlineCloudHostRequirements これらはアセット内に保存され、DeadlineCloudRenderStepアセットによって参照されます。

要件タイプ	設定場所	ユースケース
基本システム要件	CPU/RAM/GPU フィールド	特定のハードウェアを持つマシンへのレンダリングを制限する
カスタム金額の要件	名前 + 最小/最大値	数値リソースレベル (ライセンス、トークン、クォータ) を要求する
カスタム属性の要件	属性 + 値リスト	ステップを実行するために必要なフリートのカスタム属性

Note

目のアイコンを使用して要件を非表示にすると、MRQ 送信 UI でのみ要件が非表示になります。この要件は、ジョブの送信時も適用されます。

Important

カスタム金額要件とカスタム属性要件に使用される Name および Attribute 値は、公式の Open Job Description ドキュメント [「Open Job Specifications」](#) で定義されている有効な識別子と厳密に一致する必要があります。

ステップ 1: ホスト要件アセットを作成する

1. Unreal Engine でコンテンツブラウザを開きます。
2. 新しいアセットを作成する: 追加 → その他 → データアセット → DeadlineCloudHostRequirements。
3. アセットに などの名前を付けます MyHostRequirements。

ステップ 2: デフォルトの YAML テンプレートをロードする

1. アセットの詳細パネルで、テンプレートへのパスフィールドを見つけます。
2. デフォルトのテンプレート を選択します `Plugins/UnrealDeadlineCloudService/Content/Python/openjd_templates/host_requirements.yml`。

これにより、CPU、メモリ、GPU、OS、アーキテクチャなどの基本要件がロードされます。

Note

基本要件は削除できません。その値のみ変更できます。

ステップ 3: 基本 (システム) 要件を設定する

設定	例
オペレーティングシステム	windows
CPU アーキテクチャ	x86_64
vCPUs(最小)	16
GPUs (最小)	1

Max = 0 の場合、その要件の上限はありません。

ステップ 4: カスタム金額要件を追加する

要件が数値 (ライセンス数、同時実行数の制限など) の場合に使用されます。

1. 「カスタム金額要件」セクションを参照してください。
2. 新しいエントリを追加して名前を設定します。例: `amount.custom.license`。
3. 最小値と最大値を設定します。
 - 最小: 1。
 - 最大: 0 (0 は無制限) 。

ステップ 5: カスタム属性要件を追加する

タグとラベルに基づいてワーカーをフィルタリングするために使用されます。

1. 「カスタム属性の要件」セクションを参照してください。
2. 新しい属性名を追加します。例: `attr.custom.gpu.vendor`。
3. 一致するルールを選択します。
 - AllOf - すべての値が一致する必要があります。
 - AnyOf - 少なくとも 1 つ一致する必要があります。
4. スペースで区切られた属性値を入力します。例: `nvidia amd`。

ステップ 6: MRQ での可視性 (オプション)

各要件エントリには目のアイコンがあります。

目	行動
[Visible]	要件は MRQ 送信 UI に表示されます。
[非表示]	要件は非表示になっていますが、ジョブの送信時に含まれています

この設定は、技術的な制約を維持しながらアーティストの UI を簡素化するのに役立ちます。

ステップ 7: レンダリングステップに要件をアタッチする

1. 関連するDeadlineCloudRenderStepアセットを開きます。
2. ホスト要件フィールドを見つけます。
3. 作成したアセット (などMyHostRequirements) を割り当てます。

このステップを使用するすべての MRQ 送信で、ホスト選択ロジックが強制されるようになりました。

概要

ホスト要件アセットを使用すると、次のことができます。

- レンダリングジョブを実行するワーカーマシンを制御します。
- 最小リソースレベルを指定します。
- 属性マッチングを使用して特定のワーカープールをターゲットにします。
- テクニカルルールをそのまま維持しながら UI を簡素化します。
- 複数の MRQ ステップで設定を再利用します。

この設定により、ジョブのルーティングの一貫性が向上し、正しいハードウェアでレンダリングが行われるようになります。

詳細設定

サービスマネージドフリートとカスターマネージドフリート

サービスマネージドフリート (SMF)

サービスマネージドフリートでは、Unreal Engine とアダプターは、デフォルトのキュー環境の `deadline-cloud conda` チャンネルを使用して自動的に使用できます。この設定は、最も簡単なエクスペリエンスを提供します。

カスターマネージドフリート (CMF)

カスターマネージドフリートの場合、Unreal Engine とアダプターをワーカーホストに手動でインストールする必要があります。この設定は、より多くの制御を提供し、Perforce 統合などの追加機能をサポートします。詳細な手順については、「[カスターマネージドフリート \(CMF\) ワーカーのセットアップ](#)」を参照してください。

Unreal Engine レンダリング機能

Unreal Engine のレンダリングシステムは、以下を包括的にサポートします。

機能	説明	注意事項
映画レンダリングキュー	高品質のオフラインレンダリング	ジョブ送信との統合
シーケンサー	タイムラインベースのアニメーションシステム	自動ショット検出と処理

機能	説明	注意事項
プロジェクトプラグイン	カスタムプラグインのサポート	自動検出と包含
アセットの依存関係	コンテンツファイル管理	包括的なアセット追跡
スティッキーレンダリング	ショット間のアプリケーションの永続性	マルチショットシーケンスのパフォーマンスの向上

すべてのレンダリング機能は、Unreal Engine 統合送信者によって自動的に検出および設定されます。アダプターは適切な依存関係処理を維持し、Unreal Engine を再起動せずに効率的なマルチショットレンダリングをサポートします。

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [Unreal Engine の Deadline Cloud](#)

Foundry Nuke

Foundry Nuke は、テレビや映画のポストプロダクションに使用されるノードベースのデジタル合成および視覚効果アプリケーションです。Nuke AWS は、送信者、conda パッケージ、レンダリングパフォーマンスを向上させるアダプターを備えた Deadline Cloud (Deadline Cloud) でサポートされています。このガイドでは、Deadline Cloud with Nuke を使用してレンダリングタスクを複数のマシンに分散することで、プロジェクトをより迅速にレンダリングするためのstep-by-stepの手順を示します。

サポートの概要

Nuke は、次のコンポーネントでサポートされています。

- 送信者: シーンとアセットの自動検出を使用して Nuke から直接ジョブを送信するための統合送信者プラグイン。
- Conda パッケージ: nuke バージョン 15、16、および 17 をインストールするパッケージは、サービスマネージドフリートの Deadline Cloud conda チャンネルで使用できます。

- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows、macOS、Linux の送信者サポートと Linux のワーカーサポートは、自動パスマッピングでのみ行われます。

Nuke バージョンの互換性

次の表は、Nuke バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート
15	Windows、macOS、Linux	Linux
16	Windows、macOS、Linux	Linux
17	Windows、macOS、Linux	Linux

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルのサービスマネージドフリートで使用できる Nuke に適用される conda パッケージを示します。

OS	パッケージ	バージョン	注意事項
Linux	nuke	15	組み込み合成エンジンを含む
Linux	nuke	16	組み込み合成エンジンを含む
Linux	nuke	17	組み込み合成エンジンを含む
Linux	nuke-openjd		Nuke アダプターを含む

開始方法

Deadline Cloud で Nuke を使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。キューは、期限クラウド conda チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。
2. Deadline Cloud Submitter を使用してアーティストワークステーションに Deadline Cloud モニターと Nuke 送信者をインストールし、インストーラをモニタリングします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. 統合された送信者を使用して Nuke から直接ジョブをキューに送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

送信者を起動する

Nuke で Deadline Cloud 送信者を起動するには

Note

Nuke のサポートは、サービスマネージドフリートの Conda 環境を使用して提供されます。詳細については、[「デフォルトのcondaキュー環境」](#)を参照してください。

1. Deadline Cloud Submitter を使用してアーティストワークステーションに Deadline Cloud モニターと Nuke 送信者をインストールし、インストーラをモニタリングします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
2. Nuke を開きます。
3. アセットルートディレクトリ内に存在する依存関係を持つ Nuke スクリプトを開きます。
4. AWS Deadline を選択し、Deadline Cloud に送信を選択して送信者を起動します。
5. まだ認証されていない場合は、ログインを選択し、ブラウザウィンドウでユーザー認証情報を使用してログインします。
6. [Submit] を選択してください。

インストール

Deadline Cloud for Nuke 送信者をインストールするには、以下が必要です。

- Windows、macOS、または Linux ワークステーション。
- Nuke 14、15、16、または 17。これらのバージョンはサービスマネージドフリートの[デフォルトの conda キュー環境](#)でサポートされているため、Nuke 14 よりも Nuke 15 以降をお勧めします。サービスマネージドフリートで Nuke 14 を使用するには、ワーカーが Nuke 14 を使用できるようにする必要があります。推奨される方法は、[「アプリケーションまたはプラグインの conda パッケージを作成する」](#)に従って独自の conda パッケージを作成することです。

Deadline Cloud for Nuke 送信者をインストールするには、次の 2 つの方法があります。

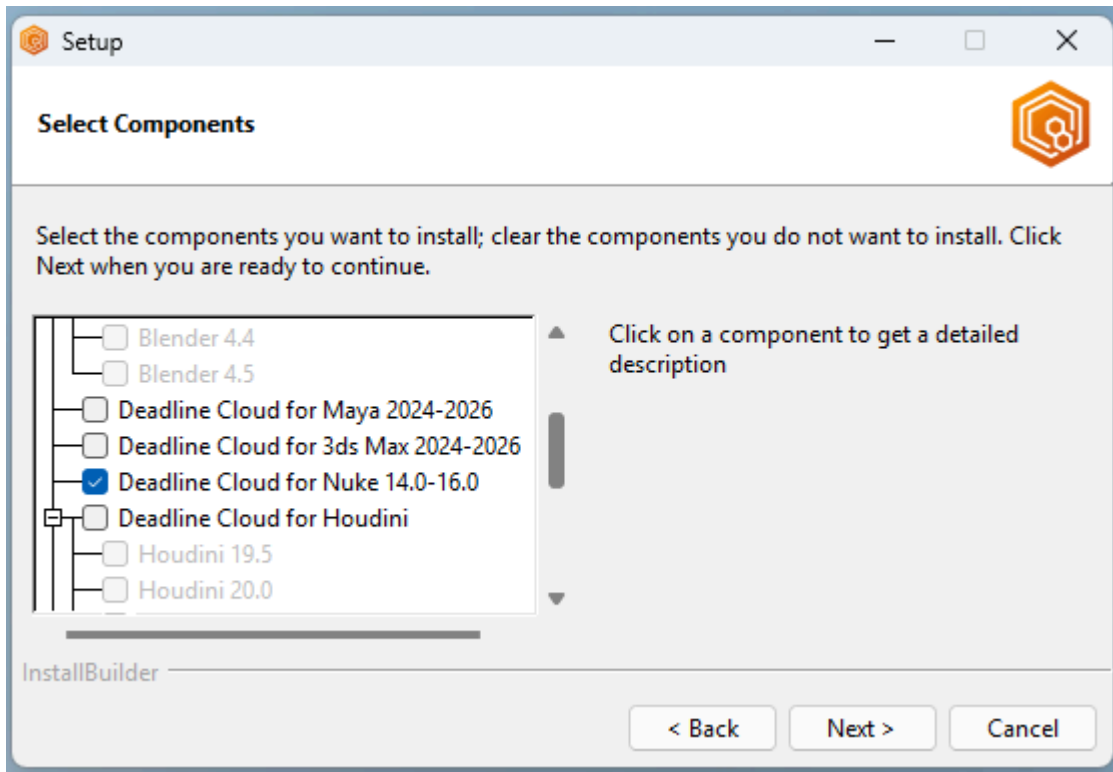
- Deadline Cloud 送信者インストーラの使用 (推奨)。
- [送信元から送信者を手動でインストールします](#)。

Deadline Cloud 送信者インストーラの使用

Deadline Cloud for Nuke 送信者は、Deadline Cloud 送信者インストーラを使用してインストールできます。

送信者をインストールするには:

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラーを実行します。
3. コンポーネントを選択するように求められたら、Nuke のチェックボックスを見つけてマークします。



4. インストーラの実行を完了します。
5. Nuke を起動します。
6. Deadline Cloud が上部のナビゲーションバーに追加されているかどうかを確認して、インストーラを確認します。

Nuke 送信者の使用

Deadline Cloud for Nuke 送信者は、次の 2 種類のジョブをサポートしています。

- レンダリングジョブ - Nuke スクリプトの 1 つ以上の [書き込みノード](#) によって作成された出力ファイルをレンダリングします。
- CopyCat トレーニングジョブ - Nuke スクリプトで [CopyCat ノード](#) のトレーニングを実行します。

ジョブのレンダリング

Deadline Cloud for Nuke 送信者を使用するには、以下が必要です。

- Deadline Cloud に送信するプロファイル。

- 送信先の Deadline Cloud ファームとキュー。

Nuke から Deadline Cloud にレンダリングジョブを送信するには:

1. Nuke ファイルを保存します。
2. 上部のナビゲーションバーから、Deadline Cloud を選択します。ドロップダウンメニューから、Deadline Cloud に送信を選択します。
3. ダイアログのタブを使用してジョブをカスタマイズします。
4. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
5. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

Nuke レンダー固有の設定

ジョブ固有の設定タブには、Nuke で作成されたジョブに固有のオプションがあります。

Submit Rendering to AWS Deadline Cloud

Shared job settings Job-specific settings Job attachments Host requirements

Write nodes All write nodes

Views All views

Override frame range 1-226

Use proxy mode

Continue on error

Use timeouts *Set a maximum duration for actions from this job*

Render task timeout 6 days 0 hours 0 minutes

Setup timeout 1 day 0 hours 0 minutes

Teardown timeout 0 days 1 hour 0 minutes

Include gizmos in job bundle

✓ myprofile

Settings... Submit Export bundle

- 書き込みノード - 出力をレンダリングする [書き込みノード](#)。すべての書き込みノードをレンダリングするか、特定のノードを選択するかを選択できます。
- ビュー - レンダリングする [ビュー](#)。

- フレーム範囲を上書きする - Nuke で設定されているものとは異なるフレームまたはフレーム範囲をレンダリングするには、このオプションを選択します。フレーム範囲は Open [Job Description](#) パターンに従います。
- プロキシモードを使用する - 送信されたジョブで [プロキシモード](#) を使用するかどうかを管理します。
- Continue on error - 選択すると、Nuke はエラーが発生したときにレンダリングを続行しようとし、オフにすると、Nuke はエラーが発生したときにタスクに失敗します。
- チャンクサイズ - 各チャンクにグループ化するフレームの数 (1 ~ 150)。タスクごとに 1 フレームに 1 を使用します (デフォルト)。値が大きいほど、フレームが連続したチャンクにグループ化され、タスクごとのオーバーヘッドが減少します。詳細については、[「ジョブテンプレートのタスクチャンキング」](#) を参照してください。
- ターゲットチャンク期間 (秒) - 値を指定すると、スケジューラは完了したチャンクの観測されたランタイムに基づいてチャンクサイズを動的に調整し、チャンクごとにこの期間を目指します。すべてのチャンクに固定チャンクサイズを使用するには、0 のままにします。
- タイムアウトを使用する - ユーザー設定のタイムアウトを使用するかどうか。
- レンダリングタスクのタイムアウト - レンダリングを実行する各アクションの最大継続時間。デフォルトは 6 日です。
- セットアップタイムアウト - シーンロードなど、レンダリング用のジョブを設定する各アクションの最大期間。デフォルトは 1 日です。
- ティアダウンタイムアウト - レンダリングに必要なセットアップをティアダウンするアクションの最大継続時間。デフォルトは 1 時間です。
- ジョブバンドルにギズモを含める - ジョブバンドルに [ギズモ](#) を含めるかどうか。

他の送信者タブの詳細については、[送信者を使用するための Deadline Cloud ガイド](#) を参照してください。

CopyCat トレーニングジョブ

Deadline Cloud for Nuke 送信者を使用して CopyCat ノードをトレーニングするには、以下が必要です。

- Deadline Cloud に送信するプロファイル。
- 送信先の Deadline Cloud ファームとキュー。

- 送信先のキューに関連付けられた GPU 対応ワーカーを持つ Deadline Cloud フリート。GPU アクセスでサービスマネージドフリートを作成する手順については、[「サービスマネージドフリートの管理」](#)を参照してください。

CopyCat トレーニングジョブを Nuke から Deadline Cloud に送信するには:

1. CopyCat ノードを含む Nuke スクリプトを作成または開きます。
2. Ground-truth ノードと入力ノードを CopyCat ノードにアタッチし、ノードのつまみを目的の値に設定します。[CopyCat の使用の詳細については、Foundry の CopyCat ドキュメント](#)を参照してください。CopyCat
3. Nuke ファイルを保存します。
4. 上部のナビゲーションバーから、Deadline Cloud を選択します。ドロップダウンメニューから、CopyCat トレーニングを Deadline Cloud に送信するを選択します。
5. ダイアログのタブを使用してジョブをカスタマイズします。
6. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
7. 送信を選択し、プロンプトに従って Deadline Cloud にジョブを送信します。

Nuke CopyCat トレーニング固有の設定

ジョブ固有の設定タブには、Nuke で作成された CopyCat トレーニングジョブに固有のオプションがあります。

Submit CopyCat Training to AWS Deadline Cloud

Shared job settings Job-specific settings Job attachments Host requirements

CopyCat Node CopyCat1

Use timeouts Set a maximum duration for actions from this job

Render task timeout 6 days 0 hours 0 minutes

Setup timeout 1 day 0 hours 0 minutes

Teardown timeout 0 days 2 hours 0 minutes

Include gizmos in job bundle

myprofile

Settings... Submit Export bundle About...

- CopyCat ノード - ノード名でトレーニングする CopyCat ノードを選択します。
- タイムアウトを使用する - ユーザー設定のタイムアウトを使用するかどうか。
- レンダリングタスクのタイムアウト - 各アクションの最大継続時間。CopyCat の場合、トレーニングは 1 つのアクションです。デフォルトは 6 日です。

- セットアップタイムアウト - シーンロードなど、ジョブを設定する各アクションの最大継続時間。デフォルトは 1 日です。
- ティアダウンタイムアウト - セットアップをティアダウンするアクションの最大継続時間。デフォルトは 1 時間です。
- ジョブバンドルにギズモを含める - ジョブバンドルに [ギズモ](#) を含めるかどうか。

他の送信者タブの詳細については、[「送信者を使用するための Deadline Cloud ガイド」](#) を参照してください。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアのバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとしています。これは、ワークステーションバージョンの Nuke が上記のバージョンテーブルに表示されない場合に失敗します。

サポートされていないバージョンの Nuke が必要な場合は、次のオプションがあります。

- Nuke からジョブを送信する場合、CondaPackages キューパラメータを上書きして、ワーカーで使用するサポートされているバージョンを指定できます (例: `nuke=17`, `nuke-openjd=*`)。これは、コンポジションで使用される機能や、Nuke がワークステーションバージョンのコンポジションとどのように連携するかに応じて、機能しない場合があります。
- ワーカーにインストールするバージョンのカスタム conda レシピとチャンネルを構築できます。以下でリンクされているサポートされているバージョンの conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#) を参照してください。

カスタム Nuke 実行可能ファイル

PATH で使用できない場合は、特定の Nuke 実行可能ファイルを指すように `NUKE_EXECUTABLE` 環境変数を設定できます。

OpenColorIO のサポート

Nuke 統合には、OpenColorIO (OCIO) 色管理ワークフローのフルサポートが含まれています。レンダーフーム全体で一貫した色処理を確保するために、色設定は自動的に検出され、ジョブの送信に含まれます。

Nuke 合成機能

Nuke の合成エンジンは、以下を包括的にサポートします。

機能	説明	注意事項
ノードの書き込み	複数の出力形式とコーデック	送信者によって自動的に検出
フレーム範囲	カスタムフレーム範囲の仕様	オーバーライド範囲とデフォルト範囲をサポート
複数のビュー	ステレオおよびマルチビューレンダリング	ビュー固有の出力の適切な処理
カラー管理	OpenColorIO 統合	OCIO 設定の自動検出
パスマッピング	クロスプラットフォームパス変換	Windows/Linux とのシームレスな互換性
CopyCat	ML ベースのペイントと回転スコープ	Nuke 14.0 以降が必要

合成機能は、Nuke 統合送信者によって自動的に検出および設定されます。送信者は、複雑な構成の適切な依存関係処理とアセット管理を維持します。

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で利用できます。

- [Nuke の Deadline Cloud](#)
- [Nuke Conda レシピ](#)は、サポートされているバージョンの GitHub で利用できます。

KeyShot Studio

KeyShot Studio は、3D モデルとアニメーションをレンダリングするために Luxion によって開発されたリアルタイムレイトレーシングおよびグローバル照明プログラムです。このガイドでは、KeyShot Studio で AWS Deadline Cloud (Deadline Cloud) を使用して、レンダリングタスクを複数のマシンに分散することでプロジェクトをより迅速にレンダリングするためのstep-by-stepの手順を示します。

サポートの概要

KeyShot Studio は、次のコンポーネントでサポートされています。

- 送信者: シーンとアセットの自動検出を使用して KeyShot から直接ジョブを送信するための統合された送信者拡張機能。
- クロスプラットフォーム互換性: Windows および macOS の送信者サポートと Windows のワーカーサポート。
- Licensing (BYOL): Bring Your Own License for KeyShot rendering on your farm.

KeyShot バージョンの互換性

次の表は、Keyshot バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	エンジンのレンダリング	ライセンス
2024	Windows、macOS	組み込みレイトレーサー	BYOL が必要
2025	Windows、macOS	組み込みレイトレーサー	BYOL が必要

前提条件

KeyShot には Bring Your Own License (BYOL) が必要です。レンダーファームフリートで使用できる有効な KeyShot ライセンスが必要です。ワーカーノードからアクセスできるようにライセンスサーバーを設定します。詳細については、[「サービスマネージドフリートをカスタムライセンスサーバーに接続する」](#)を参照してください。

サービスマネージドフリートで KeyShot を使用するには、conda パッケージを作成し、カスタム conda チャンネルでホストする必要があります。[KeyShot のサンプル conda レシピ](#)は、GitHub で入手できます。カスタム conda チャンネルの作成の詳細については、「[カスタム conda チャンネルの作成](#)」を参照してください。

開始する前に、以下の準備が整っていることを確認します。

- Windows または macOS ワークステーション。
- KeyShot Studio 2023 - 2025。
- [Deadline Cloud](#) Monitor がインストールされました。
- KeyShot Studio がインストールされ、ライセンスされているフリートを持つ Deadline Cloud ファームへのアクセス。

開始方法

Deadline Cloud で KeyShot を使用するには:

1. サーマネージドフリートを作成し、キューに関連付けます。キューは、KeyShot パッケージを含むカスタム conda チャンネルを含むキュー環境で設定する必要があります。詳細については、「[キュー環境の作成](#)」を参照してください。
2. Deadline Cloud Submitter とモニターインストーラを使用して、アーティストワークステーションに Deadline Cloud モニターと KeyShot 送信者をインストールします。詳細については、「[ワークステーションのセットアップ](#)」を参照してください。

インストール

KeyShot 送信者拡張機能を使用すると、KeyShot 内から直接 Deadline Cloud にジョブを送信できます。

送信者のインストール

送信者をインストールするには:

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラを実行し、画面の指示に従います。
3. インストール後に KeyShot を起動します。

送信者の更新

送信者を最新バージョンに更新するには、最新の送信者インストーラをダウンロードして実行します。

KeyShot 送信者の使用

シーンの準備

ジョブを送信する前に:

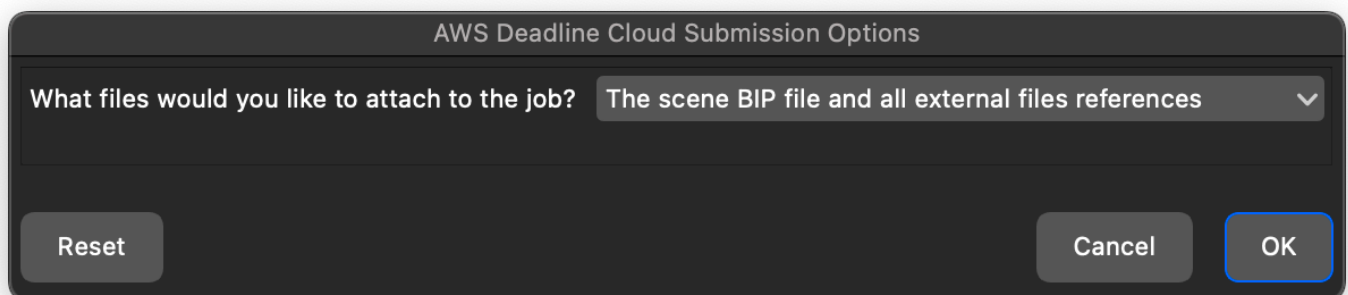
1. シーンが保存されていることを確認します。
2. 必要に応じてカメラの角度、マテリアル、照明を設定します。
3. アニメーションをレンダリングする場合は、アニメーションフレームを設定します。

ジョブの送信

1. KeyShot の上部ツールバーで、スクリプトコンソールを選択します。
2. スクリプトコンソールで、スクリプト > Deadline Cloud に送信に移動します。
3. [Run] (実行) を選択します。

送信オプション

送信者を実行すると、ファイル添付ファイルの処理方法を尋ねるダイアログが表示されます。



以下のオプションのいずれかを選択してください。

- シーン BIP ファイルとすべての外部ファイル参照 (推奨)

- シーンファイルとすべての参照ファイルを自動的にパッケージ化します。内部的には、送信者は、リンクされたすべてのファイルをバンドルし、相対パスを使用する KeyShot Package (KSP) を作成します。
- テクスチャ、モデル、その他の外部アセットがあるシーンに最適です。
- シーンをレンダリングするために必要なファイルがすべてワーカーにあることを確認します。
- シーン BIP ファイルのみ
 - KeyShot シーンファイルのみを送信します。
 - このオプションは、ワーカーが参照されるすべてのファイルへのアクセス権を既に持っている場合に使用します。
 - 外部ファイルにアクセスするには、共有ネットワークストレージまたは他の方法が必要です。

レンダリング設定

送信オプションを選択すると、Deadline Cloud 送信者インターフェイスが表示されます。レンダリング設定を構成します。

1. 共有ジョブ設定。

Submit to AWS Deadline Cloud

Shared job settings | Job-specific settings | Job attachments | Host requirements

Job Properties

Name

Description

Priority

Initial state

Maximum failed tasks count

Maximum retries per task

Maximum worker count No max worker count
 Set max worker count

Deadline Cloud settings

Farm

Queue

Queue Environment: Conda

Conda Packages

Conda Channels

Credential source:

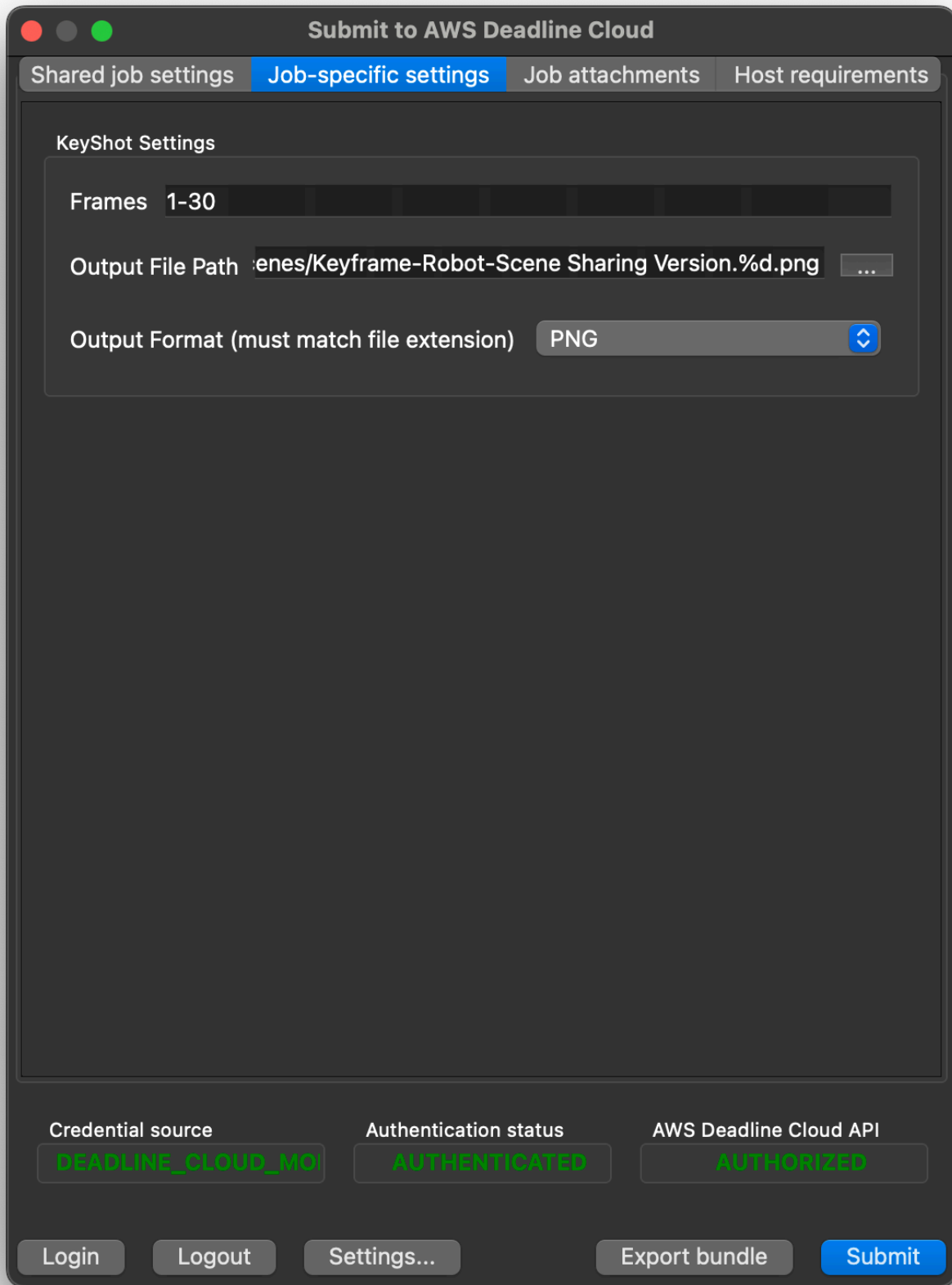
Authentication status:

AWS Deadline Cloud API:

Login Logout Settings... Export bundle **Submit**

- ジョブ名: ジョブにわかりやすい名前を付けます。
- 送信者を初めて使用する場合は、ファームとキューの設定が必要になる場合があります。設定するには、設定ボタンを選択します。

2. ジョブ固有の設定。



- フレーム: レンダリングするフレームを指定します (1-30フレーム 1~30 など)。
 - 出力ファイルパス: レンダリングされたイメージの場所と命名パターンを設定します。パスにはファイルの拡張子を含める必要があり、拡張子は出力形式と一致する必要があります。フレーム番号のプレースホルダー%dとしてを使用します。
 - 出力形式: イメージ形式 (PNG、JPEG、EXR、TIFF、PSD) を選択します。
3. ジョブアタッチメント (オプション)。ジョブにアップロードおよびアタッチするファイルを選択します。送信者はデフォルトでファイルを自動的に検出してアタッチします。
 4. ホスト要件 (オプション)。このジョブのタスクを取得できるホストのタイプを指定します。
 5. 送信を選択して、ジョブを Deadline Cloud に送信します。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはカスタム conda パッケージの KeyShot バージョンを使用します。カスタム conda チャンネルに、使用するすべての KeyShot バージョンのパッケージが含まれていることを確認します。

サポートされていないバージョンの KeyShot が必要な場合は、ワーカーにインストールする目的のバージョンのカスタム conda レシピとチャンネルを構築できます。[KeyShot のサンプル conda レシピ](#)を開始点として使用します。カスタム conda チャンネルの作成の詳細については、「[カスタム conda チャンネルの作成](#)」を参照してください。

オープンソースリソース

送信者はオープンソースであり、GitHub で利用できます。

- [KeyShot の Deadline Cloud](#)。
- [KeyShot conda レシピ](#)。
- [スタンドアロン KeyShot ジョブバンドル](#)。

Maxon シネマ 4D

Cinema 4D は、Maxon のプロフェッショナルな 3D アニメーション、モデリング、シミュレーション、レンダリングソフトウェアソリューションです。Cinema 4D AWS は、送信者、conda パッ

ページ、使用ベースのライセンス、パフォーマンスを向上させるアダプターなど、Deadline Cloud (Deadline Cloud) でサポートされています。このガイドでは、インストールから最初の正常なレンダリングまで、Cinema 4D で Deadline Cloud を使用方法について説明します。

Deadline Cloud for Cinema 4D を使用する理由は次のとおりです。

- レンダリングをスケールする - 複数のインスタンスにフレームを分散することで複雑なシーンをより速くレンダリングし、レンダリング時間を数時間から数分に短縮します。
- ワークステーションを解放する - Deadline Cloud にレンダリングを送信し、シーンがバックグラウンドでレンダリングされている間、次のプロジェクトで作業を続けます。
- 使用した分のみ支払う - 前払いコストや長期契約はありません。実際に使用したコンピューティング時間に対してのみ支払います。
- Redshift 対応 - Maxon Cinema 4D と Maxon Redshift のフルサポート。

サポートの概要

Cinema 4D は、次のコンポーネントでサポートされています。

- 送信者: シーンとアセットの自動検出機能を備えた Cinema 4D から直接ジョブを送信するための統合送信者。
- Conda パッケージ: 送信者を使用する場合のサービスマネージドフリートへの自動インストール。
- アダプター: スティックセッションと追加のモニタリングによるレンダリングをより効率的にするミドルウェア。
- クロスプラットフォーム互換性: Windows および Linux のワーカーサポートと自動パスマッピングを備えた Windows および macOS の送信者サポート。
- 使用量ベースのライセンス: Cinema 4D、Redshift、Red Giant ライセンスのPay-as-you-go。

Cinema 4D バージョンの互換性

次の表は、Cinema 4D バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート	使用状況ベースのライセンス
2024	Windows、macOS	Windows	使用状況ベースのライセンスが利用可能

メジャーバージョン	送信者のサポート	Conda サポート	使用状況ベースのライセンス
2025	Windows、macOS	Windows、Linux	使用状況ベースのライセンスが利用可能
2026	Windows、macOS	Windows、Linux	使用状況ベースのライセンスが利用可能

Deadline Cloud Conda チャンネル

次の表は、期限クラウド conda チャンネルのサービスマネージドフリートで使用できる Cinema 4D に適用されるすべての conda パッケージを示しています。

OS	パッケージ	バージョン	注意事項
Windows	cinema4d	2024	標準レンダラー、物理レンダラー、Redshift レンダラーが含まれます。
Windows、Linux	cinema4d	2025	標準レンダラー、物理レンダラー、Redshift レンダラーが含まれます。
Windows、Linux	cinema4d	2026	標準レンダラー、物理レンダラー、Redshift レンダラーが含まれます。
Windows、Linux	cinema4d-c4dtoa	2025	Cinema4D から Arnold へ
Windows	cinema4d-c4dtoa	2026	Cinema4D から Arnold へ

OS	パッケージ	バージョン	注意事項
Windows、Linux	cinema4d-openjd		Cinema 4D アダプターを含む

Note

Cinema 4D の場合、Linuxconda パッケージは素材 3D マテリアルをサポートしていません。このマテリアルを持つジョブは、次のいずれかのエラーで失敗します。

```
Commandline: ./modules/io_substance/source/substance_framework/src/details/detailsengine.cpp:794:
SubstanceAir::Details::Engine::Context::Context(SubstanceAir::Details::Engine&,
SubstanceAir::RenderCallbacks*): Assertion `res==0' failed.
```

```
/home/job-user/.conda/envs/<hash>/Lib/deadline/cinema4d_adaptor/Cinema4DAdaptor/
adaptor.sh: line 44: 10832 Segmentation fault      (core dumped) $C4DEXE
${ARGS[*]}
```

Windows 代わりに、素材マテリアルを含むジョブを に送信することをお勧めします。Cinema 4D 2025.3.3 ではLinux、グローバル化されたアセットパスがセグメンテーションの障害を引き起こす可能性があります。したがって、Linuxconda パッケージには、代わりに Redshift 2025.6.0 を含む Cinema 4D 2025.3.1 が含まれています。Cinema 4D 2025.3.3 の機能またはバグ修正が必要な場合は、Cinema 4D 2026 にアップグレードするか、Windows 代わりにそれらのジョブを に送信するという 2 つのオプションをお勧めします。Cinema 4D OpenJD では、タイムアウトの問題を防ぐために、デフォルトの 2 日間のタイムアウトを使用する代わりに、タスク実行タイムアウトを予想レンダリング時間の 2 倍に設定することをお勧めします。

開始方法

Deadline Cloud で Cinema 4D フルマネージドを使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。GPU を必要とする Redshift または Red Giant 機能を使用する場合は、GPU サポートを使用してフリートを設定します。キュー

は、deadline-cloud conda チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。

2. Deadline Cloud Submitter を使用してアーティストワークステーションに Deadline Cloud モニターと Cinema 4D 送信者をインストールし、インストーラをモニタリングします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. 統合された送信者を使用して Cinema 4D から キューにジョブを直接送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

クイックスタート

Cinema 4D と Deadline Cloud をわずか数ステップでセットアップします。

必要なもの

- Cinema 4D 2024 - 2026 がワークステーションにインストールされている。
 - Redshift、Arnold、および Cargo はネイティブでサポートされています。
- ジョブ送信用の Windows または macOS ワークステーション。
- [Deadline Cloud Monitor](#) がインストールされました。
- 次のいずれかを使用して Deadline Cloud ファームにアクセスします。
 - Windows サービスマネージャドフリート、または
 - Cinema 4D、Cinema 4D アダプター、ライセンス設定を備えたカスタマーマネージャドフリート。

ステップ 1: 送信者をインストールする

送信者は、Cinema 4D の拡張機能メニューに Deadline Cloud 機能を追加し、シーンを Deadline Cloud に直接送信してレンダリングを管理できます。

[公式インストーラ](#)をダウンロードします (推奨)。

1. インストーラを実行し、画面の指示に従います。
2. インストール後に Cinema 4D を起動します。
3. 送信者が拡張機能 > Deadline Cloud Submitter に表示されることを確認します。

送信者の更新

送信者を最新バージョンに更新するには、最新の[送信者インストーラ](#)をダウンロードして実行します。

複数のユーザー向けのシステム全体のインストール (Windows)

複数のユーザーが Cinema 4D 送信者にアクセスする必要がある共有ワークステーションまたはエンタープライズ環境では、システム全体のインストールを実行できます。

次の前提条件を満たしていることを確認する

- 管理者アカウントのアクセス。
- システムにインストールされた Cinema 4D。

インストール手順:

1. 送信者を管理者としてインストールします。
 - [Deadline Cloud 送信者インストーラ](#)を管理者として実行します。
 - インストール時に「システムインストール」オプションを選択します。
2. 初期依存関係の設定:
 - Cinema 4D を管理者として開きます (右クリック → 「管理者として実行」)。
 - 拡張機能、Deadline Cloud Submitter を選択します。
 - GUI 依存関係のインストールを求められたら、「はい」を選択します。
 - このステップでは、すべてのユーザーがインストールされたパッケージにアクセスできるようにアクセス許可を設定します。
3. 通常の使用:
 - 初期設定後、すべてのユーザーは (管理者権限なしで) Cinema 4D を正常に開くことができます。
 - Deadline Cloud 送信者はすべてのユーザーが利用できます。

アクセス許可の問題のトラブルシューティングについては、「」を参照してください[トラブルシューティング](#)。

ステップ 2: 最初のレンダリングを送信する

1. Cinema 4D を開き、シーンをロードします。

2. シーンが保存されていることを確認します。
3. 必要に応じてカメラの角度、マテリアル、照明を設定します。
4. 拡張機能、Deadline Cloud Submitter を選択します。
5. レンダリング設定を確認します。
6. [Submit] を選択してください。

ステップ 3: レンダリングをモニタリングする

まだインストールしていない場合は、上記の要件から Deadline Cloud Monitor をインストールします。

ジョブを送信したら、Deadline Cloud モニターを開いてジョブの進行状況を表示します。送信者は、フレームごとに 1 つのステップと 1 つのタスクを持つジョブを作成します。

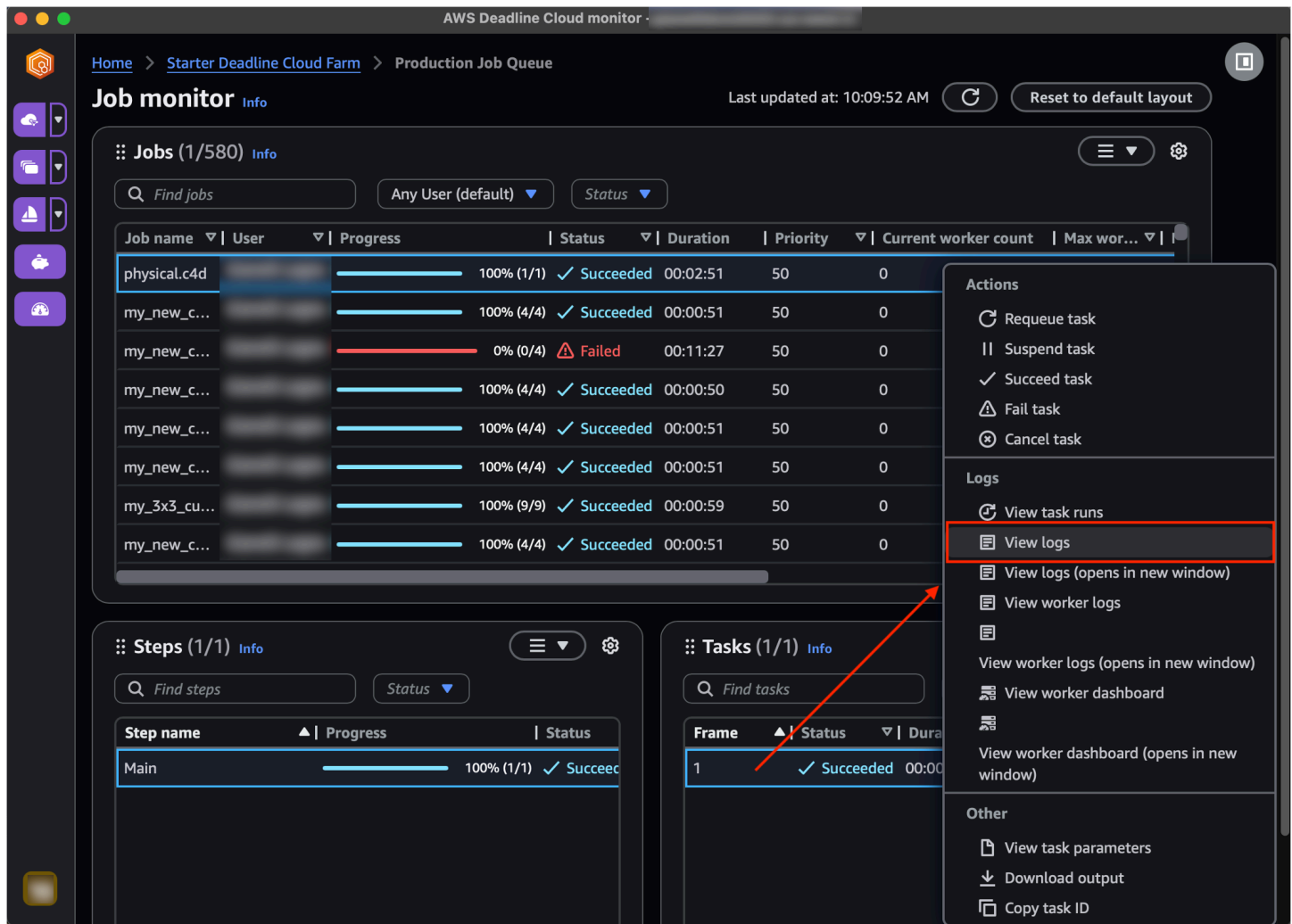
The screenshot displays the AWS Deadline Cloud monitor interface. The main section is titled "Job monitor" and shows a list of jobs. The first job, "physical.c4d", is highlighted with a red box and shows a 100% completion rate (1/1) with a "Succeeded" status. Below the job list, there are two panels: "Steps (1/1)" and "Tasks (1/1)". The "Steps" panel shows a "Main" step with a 100% completion rate (1/1) and a "Succeeded" status. The "Tasks" panel shows a task with a 100% completion rate (5/5) and a "Succeeded" status. The interface includes search bars, filters, and a sidebar with navigation icons.

Job name	User	Progress	Status	Duration	Priority	Current ...	Max wor...	Failed ta...
physical.c4d		100% (1/1)	✓ Succeeded	00:02:51	50	0	-	0
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	-	0
my_new_c...		0% (0/4)	⚠ Failed	00:11:27	50	0	-	4
my_new_c...		100% (4/4)	✓ Succeeded	00:00:50	50	0	-	0
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	-	0
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	-	0
my_3x3_cu...		100% (9/9)	✓ Succeeded	00:00:59	50	0	-	0
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	-	0

Step name	Progress	Status	Duration
Main	100% (1/1)	✓ Succeeded	00:02:52

Frame	Status	Duration	Retries /...	Start tim
1	✓ Succeeded	00:00:01	5/5	1d 4h ago

レンダリングログを表示するには、タスクのコンテキスト (右クリック) メニューを開き、ログの表示を選択します。ログの表示は、失敗したジョブのトラブルシューティングに特に役立ちます。



The screenshot shows the AWS Deadline Cloud monitor interface. The main section is titled "Job monitor" and displays a list of jobs. The table below shows the job details:

Job name	User	Progress	Status	Duration	Priority	Current worker count	Max wor...
physical.c4d		100% (1/1)	✓ Succeeded	00:02:51	50	0	
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	
my_new_c...		0% (0/4)	⚠ Failed	00:11:27	50	0	
my_new_c...		100% (4/4)	✓ Succeeded	00:00:50	50	0	
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	
my_3x3_cu...		100% (9/9)	✓ Succeeded	00:00:59	50	0	
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50	0	

A context menu is open over the failed job, showing the following options:

- Actions
 - 🔄 Requeue task
 - ⏸ Suspend task
 - ✓ Succeed task
 - ⚠ Fail task
 - ⊗ Cancel task
- Logs
 - 🔄 View task runs
 - 📄 View logs**
 - 📄 View logs (opens in new window)
 - 📄 View worker logs
 - 📄 View worker logs (opens in new window)
 - 📄 View worker dashboard
 - 📄 View worker dashboard (opens in new window)
- Other
 - 📄 View task parameters
 - 📄 Download output
 - 📄 Copy task ID

ステップ 4: 結果をダウンロードする

レンダリングジョブが正常に完了したら、レンダリングされたフレームをダウンロードできます。

1. Deadline Cloud Monitor で、完了したジョブを見つけます。
2. ジョブのコンテキスト (右クリック) メニューを開きます。
3. 出力のダウンロードを選択します。
4. レンダリングしたファイルを保存する場所を選択します。
5. ダウンロードが自動的に開始されます。

The screenshot displays the AWS Deadline Cloud monitor interface. At the top, it shows the navigation path: Home > Starter Deadline Cloud Farm > Production Job Queue. The main section is titled 'Job monitor' and shows a list of jobs. A context menu is open over a job, listing various actions and properties. The 'Download output' option is highlighted with a red box.

Job name	User	Progress	Status	Duration	Priority	Current worker count	Max wor...
physical_c4d		100% (1/1)	✓ Succeeded	00:02:51	50		
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50		
my_new_c...		0% (0/4)	⚠ Failed	00:11:27	50		
my_new_c...		100% (4/4)	✓ Succeeded	00:00:50	50		
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50		
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50		
my_3x3_cu...		100% (9/9)	✓ Succeeded	00:00:59	50		
my_new_c...		100% (4/4)	✓ Succeeded	00:00:51	50		

レンダリングされたフレームは、出力設定で指定されたのと同じ構造に編成されます。

送信者機能

Cinema 4D 送信者は、レンダリングワークフローの自動化と設定オプションを提供します。

主な利点

- スマートアセット検出 - シーンに必要なすべてのテクスチャ、モデル、その他のファイルを自動的に検索して含めます。欠落しているアセットエラーや手動ファイルハンティングがなくなりました。
- 詳細設定 - 基本設定以外の追加の設定オプションを提供し、ワークフローに合わせて出力パス、フレーム範囲、テイク、エラーチェックをカスタマイズできます。

共有ジョブ設定

ジョブ全体に適用される設定:

- ファームの選択 - ジョブがレンダリングするファームを選択します。
- キューの選択 - 選択したファーム内の特定のキューを選択します。
- ジョブ名 - レンダリングジョブにわかりやすい名前を付けます。
- ジョブの説明 - レンダリングジョブに関するオプションの詳細を追加します。
- Priority - キュー管理のジョブ優先度を設定します。
- 初期状態 - ジョブをすぐに開始するか、一時停止したままかを制御します。
- 最大失敗タスク数 - ジョブが失敗としてマークされるまでに失敗する可能性があるタスクの最大数。
- タスクあたりの最大再試行回数 - 失敗したタスクが再試行される回数。
- 最大ワーカー数 - このジョブで同時に作業できるワーカーの最大数。
- Conda パッケージ - レンダリングに必要な追加の conda パッケージを指定します。
- Conda Channels - パッケージのインストール用にカスタム conda チャンネルを定義します。

Submit to AWS Deadline Cloud ✕

Shared job settings | Job-specific settings | Job attachments | Host requirements

Job Properties

Name	<input type="text" value="physical.c4d"/>
Description	<input type="text"/>
Priority	<input type="text" value="50"/>
Initial state	<input type="text" value="READY"/>
Maximum failed tasks count	<input type="text" value="20"/>
Maximum retries per task	<input type="text" value="5"/>
Maximum worker count	<input checked="" type="radio"/> No max worker count <input type="radio"/> Set max worker count

Deadline Cloud settings

Farm	<input type="text"/>
Queue	<input type="text"/>

Queue Environment: Conda

Conda Packages	<input type="text" value="cinema4d=2024.* cinema4d-openjd=0.7.*"/>
Conda Channels	<input type="text" value="deadline-cloud"/>

Credential source: **DEADLINE_CLOUD_MONITOR_LOGIN** | Authentication status: **AUTHENTICATED** | AWS Deadline Cloud API: **AUTHORIZED**

ジョブ固有の設定

Cinema 4D レンダリングに固有の設定:

- 出力パスの上書き - メインレンダリング出力パスをシーン設定から上書きします。
- マルチパスパスの上書き - マルチパス出力パスを上書きして、追加のレンダーパスを行います。
- Takes - Cinema 4D がレンダリングする対象を選択します。
- フレーム範囲の上書き - シーン設定からフレーム範囲を上書きします。
- 自動エラーチェック - レンダリング中のエラーチェックを有効または無効にするオプションのチェックボックス。
- 詳細ログ記録を有効にする - 詳細ログ記録を有効にして、レンダリングの問題をデバッグするための詳細ログをキャプチャします。有効にすると、デバッグログが自動的にキャプチャされ、簡単にレビューできるようにジョブ出力に出力されます。
- タスク実行タイムアウト - 各タスクが完了するまでの最大時間。
- Cinema 4D 起動タイムアウト - Cinema 4D の起動に許可される最大時間。
- Cinema 4D Shutdown Timeout - Cinema 4D がクリーンにシャットダウンできる最大時間。
- アセットで Cinema 4D プロジェクトを保存する - すべてのアセットを含むプロジェクトの一時コピーを作成し、送信前にファイルパスを修正することで、レンダリング中の欠落ファイルエラーを防止します。より多くのディスク容量と送信時間を使用します。
- レンダリング中にキャッシュされたテキストを使用する - 各フレームのキャッシュされたフォントを使用してワーカーのテキストを再アニメーションすることで、テキストの誤りや欠落を防止します。レンダリング時間が長くなります。
- タイルレンダリング - 各フレームを複数のワーカー間で並列にレンダリングするタイルのグリッドに分割し、自動的に最終イメージにアセンブルします。列と行の数を設定します (それぞれ 1~99、デフォルトは 2x2)。詳細については、[タイルレンダリング](#) 以下を参照してください。
- チャンクあたりのフレーム - 各チャンクにグループ化するフレームの数 (1~150)。タスクごとに 1 フレームに 1 を使用します (デフォルト)。値を大きくすると、タスクごとのオーバーヘッドが減少します。ターゲットチャンク期間を設定すると、この値は最初のチャンクサイズとしてのみ機能します。詳細については、「[ジョブテンプレートのタスクチャンキング](#)」を参照してください。
- ターゲットチャンク期間 — チャンクあたりのターゲットレンダリング時間を秒単位で表します。Deadline Cloud は、このターゲットに到達するためにグループ化するフレーム数を自動的に調整します。常に frames-per-chunk 値を使用するには、これを 0 に設定します。

Deadline Cloud Cinema4D Submitter 0.10.0

Shared job settings | **Job-specific settings** | Job attachments | Host requirements

Override Output Path ...

Override Multi-Pass Path ...

Takes

Override Frame Range

Activate automatic error checking

Activate detailed logging

Timeouts

<input checked="" type="checkbox"/> Task Run	<input type="text" value="2 days"/>	<input type="text" value="0 hours"/>	<input type="text" value="0 minutes"/>
<input checked="" type="checkbox"/> Cinema 4D launch	<input type="text" value="0 days"/>	<input type="text" value="0 hours"/>	<input type="text" value="10 minute"/>
<input checked="" type="checkbox"/> Cinema 4D shutdown	<input type="text" value="0 days"/>	<input type="text" value="0 hours"/>	<input type="text" value="5 minutes"/>

Cinema 4D submission options

Save Cinema 4D project with assets before submission

Prevents missing file errors during rendering by creating a temporary copy of your project with all assets and fixing file paths before submission. Uses more disk space and submission time.

Cinema 4D rendering options

Use cached text during render

Prevents incorrect or missing text by using cached fonts. If there are no fonts in the scene, this is ignored. If there are fonts in the scene, this will increase rendering time.

Tile Rendering

Enable Tile Rendering

Columns

Rows

送信者機能 Version latest 234

オプションのタブ

- ジョブアタッチメント (オプション) - ジョブにアップロードしてアタッチするファイルを選択します。ファイルはデフォルトで自動的に検出され、アタッチされます。
- ホスト要件 (オプション) - このジョブのタスクの取得の対象となるホストのタイプを指定できます。

送信者は技術的な詳細を処理するため、クリエイティブな作業に集中できます。

タイルレンダリング

タイルレンダリングは、各フレームを複数のワーカー間で個別にレンダリングする小さなタイルのグリッドに分割し、最終的なフル解像度イメージに自動的にアセンブルします。タイルレンダリングは、単一フレームのレンダリングに時間がかかる大規模または複雑な単一フレームシーンに役立ちます。

有効化する方法

1. ジョブ固有の設定タブで、タイルレンダリンググループを見つけます。
2. タイルレンダリングを有効にするを選択します。
3. タイルグリッドの列数と行数を設定します (デフォルト: 2x2)。

仕組み

有効にすると、送信者はテイクごとに 2 ステップのジョブを作成します。

1. レンダリングステップ - 各タイルは個別のタスクです。3x3 グリッドは、フレームごとに 9 つのタイルタスクを生成し、それぞれが割り当てられたタイルのみをレンダリングします。
2. アセンブリステップ - フレームのすべてのタイルが終了すると、アセンブリタスクは自動的にそれらを最終的なフル解像度イメージにステッチします。ビューティ出力とマルチパス出力の両方がアセンブルされます。

手動ステッチングや外部ツールは必要ありません。

出力のダウンロード

レンダリングステップは、アセンブリの入力として使用される中間タイルイメージファイル (、など `image_0_tile_1_0.png`) `image_0_tile_0_0.png` を生成します。最終的な複合イメージのみ

が必要な場合は、レンダリングステップではなくタイルの組み立てステップから出力をダウンロードします。Assemble Tiles ステップには、すべてのタイルが結合された最終的なフル解像度イメージのみが含まれます。

デバッグの詳細ログ記録

詳細ログのアクティブ化機能は、詳細ログをキャプチャしてレンダリングの問題をトラブルシューティングするのに役立ちます。

詳細ログ記録を使用するタイミング

次の必要がある場合は、詳細ログ記録を有効にします。

- Redshift のレンダリングの問題や予期しない動作をデバッグします。
- レンダリングアーティファクトまたはエラーを調査します。
- レンダラーのパフォーマンスと動作を分析します。
- テクニカルサポートの詳細情報を取得します。

仕組み

ジョブ固有の設定で「詳細ログ記録を有効にする」チェックボックスを有効にすると、次のようになります。

1. ログキャプチャ - REDSHIFT_DEBUGCAPTURE環境変数を設定することで、システムによって Redshift デバッグログ記録が自動的に有効になります。
2. ログ出力 - レンダリングが完了すると、すべての Redshift ログがジョブ出力に自動的に出力され、簡単に確認できます。

ログの表示

詳細なログを表示するには:

1. Deadline Cloud Monitor を開きます。
2. 完了したジョブに移動します。
3. タスクのコンテキスト (右クリック) メニューを開き、ログの表示を選択します。
4. すべてのタスクのログを表示ボタンを有効にします。
5. タスク実行ログをスクロールして、詳細なログのDetailedLoggingのシャットダウン」セクションを見つけます。

Redshift ログは HTML 形式で出力され、各行の先頭にタイムスタンプが含まれます。タイムスタンプなしでウェブブラウザにログを保存して表示する場合は、提供されているクリーンアップユーティリティを使用できます。

1. Deadline Cloud モニターから詳細なログファイルをダウンロードします。タスクのコンテキスト (右クリック) メニューを開き、ログのダウンロードを選択します。
2. ログファイルパスを使用してクリーンアップスクリプトを実行します。

```
cd deadline-cloud-for-cinema-4d/scripts
python clean_redshift_detailed_logs.py /path/to/detailed_logs.log
```

または、引数なしで実行し、プロンプトが表示されたらパスを入力します。

```
python clean_redshift_detailed_logs.py
```

3. 生成された `redshift_log_cleaned.html` ファイルをウェブブラウザで開きます。

クリーンアップスクリプトは、詳細なログファイルから Redshift HTML ログを自動的に抽出し、各行からタイムスタンププレフィックス (など `2024/11/17 14:23:45-08:00`) を削除して、ログの読み取りと比較を容易にします。

重要な注意事項

- オーバーヘッドを最小限に抑えるため、詳細ログ記録はデフォルトで無効になっています。
- 特定の問題をデバッグする必要がある場合にのみ有効にします。
- ログはジョブごとにキャプチャされます。各ジョブには独自のログ出力があります。
- この機能は、Windows および Linux ワーカーノードで動作します。

トラブルシューティング

以下のセクションでは、Deadline Cloud で Cinema 4D を使用する際に発生する可能性がある一般的な問題とその解決方法について説明します。

質問のレンダリング

Q: タイルは何個使用すればよいですか？

A: シーンによって異なります。3x3 または 4x4 グリッドが良い出発点です。タイルが多いほど、並列処理は増えますが、タスクとオーバーヘッドも増えます。フレームあたりの合計タスク数は (列 x 行) + アセンブリの場合は 1 です。Deadline Cloud には、ステップごとに最大 10,000 個のタスクがあります。この制限を超えると、ジョブは CREATE_FAILED ステータスで失敗します。たとえば、1 つのフレームで 99x99 グリッドを使用すると、9,801 個のレンダリングタスクが生成され、これは制限に近くなります。

Q: macOS で送信中にフォントが機能しないのはなぜですか？

A: フォント機能は、技術的な制限により、現在 Windows でのみサポートされています。これは、Maxon の公式ドキュメントで確認されているように、混合 macOS/Windows 環境での既知の動作です。詳細については、「[Maxon's official FAQ on resolve missing fonts in Team Render](#)」を参照してください。

一般的な問題

Q: 送信ボタンが Cinema 4D に表示されません。

A: 拡張機能が正しくインストールされ、Cinema 4D が再起動されていることを確認します。コンソール (拡張機能 > コンソール) でエラーメッセージを確認します。問題が解決しない場合は、送信者を再インストールすることを検討してください。

Q: ジョブは正常に完了しましたが、出力をダウンロードしようとするとき「出力が見つかりません」という結果になります。なぜですか？

A: これは、ほとんどの場合、Cinema 4D 送信者と Deadline Cloud モニターのバージョンが一致しません。0.11.1 (以降) deadline-cloud-for-cinema-4d で送信されたジョブは Deadline Cloud Monitor 1.1.7 以前と互換性がないため、レンダリング自体が成功しても出力のダウンロードが失敗します。

影響を受けることを確認する方法:

1. 送信者のバージョンを確認します。バージョンは送信者ウィンドウのタイトルに表示されます。バージョンが 0.11.1 以降の場合は、新しい送信者が存在します。
2. Deadline Cloud モニターのバージョンを確認します。Deadline Cloud Monitor を開き、About ダイアログを表示します (macOS では Deadline Cloud Monitor メニュー > Deadline Cloud Monitor について、Windows では Help メニュー > Deadline Cloud Monitor について)。バージョンが 1.1.7 以前の場合は、古い Deadline Cloud モニターがあります。

両方の条件が true の場合、この問題が発生しています。

修正: Deadline Cloud モニターを 1.1.8 以降に更新しました。[Deadline Cloud Monitor のダウンロードページから最新バージョンをダウンロードできます](#)。更新後、Deadline Cloud モニターでジョブを再度開くと、出力は期待どおりにダウンロードされます。再送信は必要ありません。

Q: レンダリングされた出力にテクスチャやアセットの一部がないのはなぜですか？

A: これは通常、パスマッピングの問題です。Cinema 4D は、編集できないシーンファイル内のアセットへのディープリンク (絶対パス) を保存することがあります。シーンがファームで送信およびレンダリングされても、アセットがアップロードされても、元のワークステーションパスを参照する場合があります。

回避策: 送信者のジョブ固有の設定タブで「アセットで Cinema 4D プロジェクトを保存」を有効にします。この設定では、すべてのアセットをプロジェクトフォルダに統合し、送信前にパスを修正して、ファームで正しくレンダリングされるようにします。

Q: Deadline Cloud に Cinema 4D ジョブを送信すると、ネストされた Redshift プロキシファイルが検出されますか？

A: いいえ。Cinema 4D ジョブを Deadline Cloud に送信しても、Redshift プロキシファイルは検出されません。これは Redshift *.rs ファイル形式の制限です。RS Proxy オブジェクトを含む Cinema 4D シーンをエクスポートすると *.rs、参照されるすべてのプロキシデータが 1 つのファイルにフラット化またはインライン化されます。外部参照は保持されません。Cinema 4D SDK は、ネストされた依存関係を検出するために *.rs ファイルを読み取ることはできず、Redshift Core もこの機能を公開しません。詳細については、[Maxon 開発者フォーラムの投稿](#)を参照してください。

Q: Windows でのシステム全体のインストールでアクセス許可エラーが発生します。これを修正するにはどうすればよいですか？

A: システム全体のインストール後に送信者にアクセスする際にアクセス許可エラーが発生した場合:

1. 初期設定が完了したことを確認します。
 - Cinema 4D が管理者として少なくとも 1 回開かれたことを確認します。
 - 依存関係のインストールプロンプトが受け入れられたことを確認します。
 - インストールがエラーなしで完了したことを確認します。
2. ファイルのアクセス許可を確認します。
 - インストールディレクトリに移動します (例: C:\Program Files \DeadlineCloudSubmitter\).
 - 右クリック → プロパティ → セキュリティタブ。

- 「ユーザー」グループに「読み取りと実行」のアクセス許可があることを確認します。
 - アクセス許可は、すべてのサブディレクトリとファイルに継承する必要があります。
3. 手動アクセス許可の修正 (必要な場合):
- コマンドプロンプトを管理者として開きます。
 - 実行: `icacls "C:\Program Files\DeadlineCloudSubmitter" /grant *S-1-5-32-545:(OI)(CI)(RX) /T。`
 - これにより、すべてのユーザーに読み取りおよび実行のアクセス許可が付与されます。
4. Cinema 4D Python 環境を確認します。
- 影響を受けるユーザーとして Cinema 4D を開きます。
 - 拡張機能、コンソールを選択します。
 - インポートを試してください: `import deadline。`
 - これが失敗すると、アクセス許可が正しく適用されない可能性があります。

サポート情報

このセクションでは、トラブルシューティングの手順と、必要なときにサポートを受ける方法について説明します。

サポートに連絡する前に

サポートを受ける前に、以下のトラブルシューティング手順を試してください。これらは多くの場合、一般的な問題を解決し、サポートに連絡する必要がある場合により良い情報を提供するのに役立ちます。

トラブルシューティングチェックリスト:

- 1つのフレームをローカルにレンダリングする - クラウドに送信する前に、Cinema 4D で少なくとも1つのフレームをローカルにレンダリングして、シーンが正しくレンダリングされることを確認します。このステップは、シーン固有の問題とクラウドレンダリングの問題を特定するのに役立ちます。
- 最新の送信者への更新 - バグ修正と改善により、頻繁に更新をリリースします。問題は新しいバージョンで既に修正されている可能性があります。最新バージョンが実行されているかどうかを確認するには:
 - 現在のバージョンを検索する: バージョンは送信者ウィンドウのタイトルに表示されます。
 - 最新リリースと比較する: [リリースページ](#)にアクセスして最新バージョンを確認します。

- バージョンが古い場合は、送信者を更新し、問題を報告する前にもう一度テストしてください。
- ジョブ固有の設定を確認する - 送信者のジョブ固有の設定タブを確認します。特に、アセットで Cinema 4D プロジェクトを保存 チェックボックスを有効にします。この設定では、すべてのアセットを含むプロジェクトの一時コピーを作成し、ファイルパスを修正します。これにより、欠落しているファイルを特定し、レンダーファームのアセットを整理できます。詳細については「[ジョブ固有の設定](#)」を参照してください。
- さまざまな Cinema 4D バージョンを試す - 問題が発生した場合は、Cinema 4D 2024、2025、または 2026 でテストして、問題がバージョン固有かどうかを確認します。
- 既存の GitHub の問題を確認する - [GitHub の問題ページ](#)を検索して、他のユーザーが既に問題を報告して解決策を見つけたかどうかを確認します。
- 別のフリートオペレーティングシステムを試す - 可能であれば、Windows フリートと Linux フリートの両方にジョブを送信します。Windows では一般的に、Cinema 4D 機能のサポートと互換性が向上します。
- セッションログの確認 - Deadline Cloud モニターからセッションログをダウンロードして確認します。これらのログには、通常、問題を特定するエラーメッセージが含まれていることがよくあります。
- シーンプロジェクトファイルを作成する - Cinema 4D の File > Save Project with Assets を使用して、すべての依存関係を含む自己完結型プロジェクトを作成します。サポートと簡単に共有できるように、このファイルを圧縮します。

サポートに連絡するタイミング

さまざまなタイプの問題は、さまざまなサポートチャンネルに転送する必要があります。

AWS の一般的なサポート

AWS サポートにお問い合わせください。

- AWS アカウントの問題。
- 請求に関する質問。
- AWS サービスの一般的な質問。
- Deadline Cloud モニターまたは CLI 使用中の Deadline Cloud 内部サーバーエラー。

AWS サポートを通じて Cinema 4D 送信者またはアダプターの問題を報告することもできますが、これらのリクエストは統合リポジトリのメンテナーにルーティングする必要があるため、時間がか

かる場合があることに注意してください。Cinema 4D-specific問題に迅速に対応するために、GitHubの問題を使用することをお勧めします。

[AWS サポートにお問い合わせください。](#)

Cinema 4D 送信者またはアダプターのサポート

Cinema 4D-specific問題のプライマリチャンネルとして GitHub の問題を使用します。

- 送信者のバグまたはクラッシュ。
- アダプターの問題。
- Cinema 4D に固有のレンダリングエラー。
- 機能リクエスト。
- 統合の問題。

[GitHub の問題を開きます。](#)

GitHub で問題を報告する方法

バグレポート

新しいバグレポートを作成する前に:

1. [既存のバグを検索](#)して、問題が最新バージョンで既に報告または修正されているかどうかを確認します。
2. 問題に一致する既存の問題が見つかった場合:
 - 優先順位付けに役立つサムズアップリアクションを追加します。
 - 提供できる追加の詳細または再現手順についてコメントします。
 - これにより、影響を受けるユーザーの数を把握できます。

既存の問題が一致しない場合は、[バグレポートテンプレートを使用して新しいバグレポートを作成します](#)。テンプレートは、必要な情報をすべて提供する手順を示します。

機能のリクエスト

新機能リクエストを作成する前に:

1. [既存の機能強化を検索](#)して、誰かが既にアイデアを提案しているかどうかを確認します。
2. 一致する既存のリクエストが見つかった場合:

- サムズアップリアクションを追加して、サポートを表示します。
- 特定のユースケースにコメントする - これにより、リクエストが強化され、さまざまなニーズを理解するのに役立ちます。
- 関心を表現するユーザーが多いほど、優先度が高くなります。

既存のリクエストが一致しない場合は、[機能リクエストテンプレートを使用して新しい機能リクエストを作成します](#)。テンプレートは、必要な情報をすべて提供する手順を示します。

Note

機能リクエストは開発の優先順位付けに役立ちますが、実装タイムラインの保証はありません。

サポートリクエストに含める内容

必要な情報チェックリスト:

サポートに連絡するとき、または GitHub の問題を作成するときは、常に以下を含めてください。

- Cinema 4D バージョン - Cinema 4D 2025.1.0 など。
- オペレーティングシステム - Windows 11、macOS 14.2 など。
- 送信者バージョン - About パネルの内容全体をコピーします (拡張機能 > Deadline Cloud Submitter > About)。
- レンダラー - スタンダード、Redshift、Arnold など。
- フリート設定 - ワーカー OS (Windows または Linux)、メモリ要件、ディスク容量、Redshift を使用している場合の GPU タイプ。
- エラーメッセージ - パラフレーズではなく、エラーテキストを入力します。

ログファイルを収集する方法

ログは問題の診断に不可欠です。詳細ログ記録を有効にするには:

1. Cinema 4D 送信者で、ジョブ固有の設定で詳細ログを有効にするを選択します。
2. ジョブを送信します。
3. ジョブが完了 (または失敗) したら、ログを取得します。

- Deadline Cloud Monitor を開きます。
- ジョブに移動します。
- ログのダウンロードを選択し、セッション全体を選択して、サポートと共有するための完全なセッションログをダウンロードします。

シーンプロジェクトファイルを作成する方法

シーンプロジェクトファイルは、Cinema 4D シーンとそのすべてのアセットをパッケージ化します。

1. まず、保存する前にシーンから機密のアセットやデータを削除します。
2. Cinema 4D で、File > Save Project with Assets に移動します。
3. 送信先フォルダを選択します。
4. Cinema 4D は、シーンと参照されるすべてのアセットをこのフォルダにコピーします。
5. プロジェクトフォルダ全体を圧縮します。
6. zip ファイルを サポートと共有します。

Important

「Save Project with Assets」を使用する前に、機密アセットを削除します。保存後にアセットを削除すると、ファイル不足エラーが発生し、元の問題の診断が困難になる可能性があります。

ヒント: 再現可能なテストケースを共有する最善の方法は、公開されているシーンや、機密資料なしで問題を示す簡略化されたシーンを使用することです。非専有アセットで問題を再作成できれば、サポートによる調査がはるかに簡単になります。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョンをインストールしようとします。Cinema 4D のワークステーションバージョンが上記のバージョン表に表示されない場合、これは失敗します。

サポートされていないバージョンの Cinema 4D が必要な場合は、ワーカーにインストールするバージョンのカスタム conda レシピとチャンネルを構築できます。以下のオープンソースリソースセクションにリンクされているサポートされているバージョンの conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

Cinema 4D の別のバージョン用に conda パッケージを作成する場合は、ライセンスが正しく取得されることを確認する必要があります。バージョンが上記の表でサポートされているバージョンのライセンスと互換性がある場合、使用量ベースのライセンスは自動的に機能します。サービスマネージドフリートを[カスタムライセンスサーバーに接続すると、独自のライセンスをサービスマネージドフリートに持ち込むこともできます。](#)

Cinema 4D プラグイン

プラグイン	プラグインバージョン	Conda レシピの提供	提供されている SMF Conda パッケージ	使用状況ベースのライセンスサポート
Redshift	2026 年 3 月 0 日	バンドル済み*	はい	はい
Redshift	2025 年 6 月 0 日	バンドル済み*	はい	はい
Red Giant	2025.x	いいえ	なし	はい
V-Ray	7.x	はい	いいえ	はい
Insydium X パーティクル	2024.x	はい	いいえ	該当なし
C4DtoArnold	4.8.4.1	はい	はい	はい

*基本 Cinema 4D パッケージレシピに含まれています

Maxon Redshift

Redshift レンダラーはすべての Cinema 4D conda パッケージに含まれており、Cinema 4D 統合送信者を使用する場合、必要に応じて自動的に使用されます。レンダリングに Redshift を使用する場

合、追加のライセンス料金が適用されます。Deadline Cloud の料金の詳細については、[「Deadline Cloud の料金」](#)を参照してください。

Maxon Red Giant

Red Giant は、ビデオのポストプロダクション、モーショングラフィックス、ビジュアルエフェクト用に設計された包括的なツールキットです。リッチなカラーグレーディング、スムーズな遷移、リアルなビジュアルエフェクト、モーシオンデザインテンプレート、ビジュアルを作成および編集するためのツールを提供します。詳細については、[「Red Giant」](#)を参照してください。

Red Giant では、サービスマネージドフリートでカスタムセットアップが必要です。Deadline Cloud フリートで使用できるホスト設定スクリプトが用意されています。一度設定すると、Red Giant は Deadline Cloud Usage-based Licensing でサポートされ、動作にそれ以上の設定は必要ありません。

V-Ray プラグイン

V-Ray は、3D フォトリアリスティックなレイトレーレンダリングプラグインです。V-Ray for Cinema 4D は現在、サービスマネージドフリートでは完全にはサポートされていません。Conda レシピが用意されており、これを使用して Deadline Cloud ファームで使用する独自の Conda チャンネルを作成できます。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。インストール後、V-Ray は Deadline Cloud Usage-based Licensing でサポートされ、動作にそれ以上の設定は必要ありません。

C4DToArnold

Autodesk Arnold ソフトウェアは、高度な Monte Carlo レイレーシングレンダラーです。詳細については、[「Arnold」](#)を参照してください。C4DToArnold は、サービスマネージドフリートで完全にサポートされています。

Insydium X パーティクル

X パーティクルは、Maxon の Cinema 4D 用のフル機能のアドバンストパーティクルおよび VFX システムです。詳細については、[「X パーティクル」](#)を参照してください。Insydium X パーティクルは現在、サービスマネージドフリートでは完全にはサポートされていません。Conda レシピが用意されており、これを使用して Deadline Cloud ファームで使用する独自の Conda チャンネルを作成できます。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。X-Particles パッケージから conda パッケージを作成すると、購入したライセンスが含まれます。サービスマネージドフリートで運用するために追加の設定は必要ありません。

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。

- [Deadline Cloud for Cinema 4D](#)
- [Cinema 4D Conda レシピ](#)は、GitHub for C4D 2024、C4D 2025、INSYDIUM X-PARTICLES プラグイン、C4DtoA プラグイン、および V-Ray プラグインで利用できます。
- Red Giant プラグインをサポートする[ホスト設定スクリプト](#)が含まれています。

SideFX Houdini

SideFX Houdini は、映画、テレビ、広告、ビデオゲームパイプラインでモデリング、リギング、アニメーション、VFX、ルック開発、ライティング、レンダリングを行うための 3D 手続き型ソフトウェアです。Houdini は Deadline Cloud で完全にサポートされており、送信者、conda パッケージ、レンダリングパフォーマンスを向上させるアダプターなどの包括的な統合が可能です。このガイドでは、Deadline Cloud with Houdini AWS を使用してレンダリングタスクを複数のマシンに分散することで、プロジェクトをより迅速にレンダリングするためのstep-by-stepの手順を示します。

サポートの概要

Houdini は、次のコンポーネントでサポートされています。

- 送信者: Houdini からジョブを直接送信するための統合レンダリング出力ノード (ROP)。シーンとアセットの自動検出機能を備えています。
- Conda パッケージ: サービスマネージドフリートへの自動インストールの Deadline Cloud。
- アダプター: スティックセッションと追加のモニタリングによる効率的なレンダリングのためのミドルウェア。
- クロスプラットフォーム互換性: Windows、macOS、Linux の送信者サポートと、自動パスマッピングによる Windows と Linux のワーカーサポート。

Houdini バージョンの互換性

次の表は、Houdini バージョンの現在のサポートレベルを示しています。

メジャーバージョン	送信者のサポート	Conda サポート	エンジンのレンダリング	使用状況ベースのライセンス
19.0	Windows、macOS、Linux	Linux	Mantra、Karma CPU、Karma XPU	使用状況ベースのライセンスが利用可能に
19.5	Windows、macOS、Linux	Linux	Mantra、Karma CPU、Karma XPU	使用状況ベースのライセンスが利用可能に
20.0	Windows、macOS、Linux	Linux	Mantra、Karma CPU、Karma XPU	使用状況ベースのライセンスが利用可能に
20.5	Windows、macOS、Linux	Linux	Mantra、Karma CPU、Karma XPU	使用状況ベースのライセンスが利用可能に
21.0	Windows、macOS、Linux	Linux	Mantra、Karma CPU、Karma XPU	使用状況ベースのライセンスが利用可能に

Deadline Cloud Conda チャンネル

次の表に、期限クラウド conda チャンネルのサービスマネージドフリートで使用できる Houdini に適用されるすべての conda パッケージを示します。

OS	パッケージ	バージョン	注意事項
Linux	フーディーニ	19.0	Mantra および Karma レンダラーを含む
Linux	フーディーニ	19.5	Mantra および Karma レンダラーを含む

OS	パッケージ	バージョン	注意事項
Linux	フーディーニ	20.0	Mantra および Karma レンダラーを含む
Linux	フーディーニ	20.5	Mantra および Karma レンダラーを含む
Linux	フーディーニ	21.0	Mantra および Karma レンダラーを含む
Linux	houdini-openjd		Houdini アダプターを含む

開始方法

Deadline Cloud で Houdini を使用するには:

1. サービスマネージドフリートを作成し、キューに関連付けます。キューは、期限クラウド conda チャンネルをサポートするキュー環境で設定する必要があります。詳細については、[「キュー環境の作成」](#)を参照してください。
2. Deadline Cloud Submitter とモニターインストーラを使用して、アーティストワークステーションに Deadline Cloud Monitor と Houdini 送信者をインストールします。詳細については、[「ワークステーションのセットアップ」](#)を参照してください。
3. 統合された送信者を使用して Houdini から直接ジョブをキューに送信します。
4. Deadline Cloud モニターを使用してジョブをモニタリングし、出力をダウンロードします。

インストール

Deadline Cloud for Houdini 送信者をインストールするには、以下が必要です。

- Windows、macOS (arm64)、または Linux ワークステーション。
- サポートされている Houdini のバージョン。

送信者のインストール

送信者をインストールするには

1. [Deadline Cloud 送信者インストーラ](#)をダウンロードします。
2. インストーラーを実行します。
 - プロンプトが表示されたら、送信者を使用する Houdini の各バージョンを選択します。
3. Houdini を起動します。

Deadline Cloud 送信者は、レンダー出力 (ROP) ノードとして自動的に使用できます。

Note

送信者インストーラは、Windows、macOS、Linux で使用できます。手動インストールについては、[GitHub リポジトリの手動インストール手順](#)を参照してください。

送信者が正しくインストールされていることを確認する

1. Houdini を開きます。
2. ネットワークエディタで、/outネットワークを選択します。
3. コンテキストメニューを開き (右クリックまたは Tab キーを押します)、 を検索します deadline。
4. Deadline Cloud を選択して新しいノードを作成します。

Houdini 送信者の使用

Deadline Cloud for Houdini 送信者は、レンダー出力 (ROP) ノードを入力として受け入れるノードです。このノードを使用してジョブを設定して送信できます。ジョブを送信すると、各 ROP のステップがグラフに含まれます。

Houdini からのジョブの送信

Deadline Cloud for Houdini 送信者を使用するには、以下が必要です。

- Deadline Cloud に送信するプロファイル。

- 送信先の Deadline Cloud ファームとキュー。

Houdini から Deadline Cloud にジョブを送信するには

1. ネットワークエディタで、/out ネットワークを選択します。
2. コンテキストメニューを開き (右クリックまたは Tab キーを押します)、 を検索deadlineして Deadline Cloud ノードを作成します。
3. ROP の出力を Deadline Cloud ノードの入力に接続します。
 - ノードを Deadline Cloud ノードに接続すると、送信されたジョブは入力 ROP とすべての ROPs をグラフにレンダリングします。
4. Deadline Cloud ノードを選択します。
5. ノードエディタのオプションを使用してジョブを設定します。各オプションの動作については [Houdini 固有の設定](#)、「」を参照してください。
6. (オプション) ジョブの関連ファイルをジョブ履歴ディレクトリに送信せずにエクスポートするには、バンドルのエクスポートを選択します。
7. 送信を選択して、ジョブを Deadline Cloud に送信します。

Houdini 固有の設定

Deadline Cloud ノードのジョブ固有の設定タブには、Houdini ジョブに固有のオプションが表示されます。

- 依存関係を別々のステップとして送信する - ROP グラフを別々のレンダリングステップに分割して、モニタリングとデバッグを容易にします。有効にすると、接続された各レンダーノードがジョブ内の独自のステップになります。
- アダプターホイールを含める - レンダリング動作を変更するアダプター (ホイールと呼ばれます) のカスタムビルドを有効にします。有効にすると、アダプターホイールを含むディレクトリを指定できます。 [GitHub リポジトリで build_wheels.sh スクリプト](#) を実行して、アダプターホイールを構築できます。
- アダプターホイール - カスタムアダプターホイールを含むディレクトリパスを指定します (アダプターホイールを含めるが有効になっている場合にのみ使用できます)。
- ROPs する - 送信中に依存関係 ROPs を自動的にロック解除します。ロックされた ROPs 既存の出力を使用し、再レンダリングしません。これにより、依存関係の再レンダリングがブロックされる可能性があります。

- シーン (.hip) 参照を自動的に解析する - ジョブの送信中に ROP グラフに基づいて、ジョブの入出力ファイル名とディレクトリを自動的に検出してアタッチします。
- シーン (.hip) ファイルの自動保存 - ジョブの送信\$HIP時にシーン (.hip) ファイルを に自動的に保存します。

その他の送信者オプションの詳細については、[送信者を使用するための Deadline Cloud ガイド](#)を参照してください。

Deadline Cloud ジョブのレンダリング戦略の上書き

多くのタイプのノードでは、フレームを任意の順序で個別にレンダリングできます。シミュレーションなどの他の場合、各フレームは前のフレームの結果に依存し、順番にレンダリングする必要があります。送信者は、そのタイプに基づいて各ノードのレンダリング戦略を選択しますが、デフォルトを上書きすることもできます。

並列レンダリングとシーケンシャルレンダリング

並列レンダリングの場合、各フレームには独自のタスクがあり、タスクは使用可能なワーカーに分散されます。シーケンシャルレンダリングの場合、ノードのすべてのフレームは、単一のワーカーで実行される単一のタスクでレンダリングされます。

デフォルトでは、ノードがシミュレーションの初期化 OPsが有効になっているジオメトリノードである場合、順次レンダリングされます。それ以外の場合、ノードは並行してレンダリングされます。

レンダリング戦略パラメータの追加

レンダーノード (Mantra や Karma など) に SEQUENTIALまたは の値でdeadline_cloud_render_strategyパラメータを作成することで、レンダー戦略を上書きできますPARALLEL。

パラメータを追加してレンダリング戦略を上書きするには

1. /out ネットワークのノードのコンテキストメニューを開きます (右クリック)。
2. パラメータとチャンネルを選択し、パラメータインターフェイスを編集します。
3. 「パラメータの作成」、「タイプ」で「順序付きメニュー」を選択します。
4. パラメータの作成列の横にある右矢印を選択して、順序付けられたメニューを既存のパラメータに追加します。
5. 既存のパラメータで新しいパラメータを選択し、パラメータの説明でその設定を編集します。

- パラメータタブで、次の操作を行います。
 - [名前] に `deadline_cloud_render_strategy` と入力してください。
 - Label には、 と入力します `Deadline Cloud Render Strategy`。
- メニュータブで、次のメニュー項目を追加します。

トークン	ラベル
シーケンシャル	シーケンシャル
PARALLEL	並行

6. [Accept (承諾)] を選択します。

ノードのパラメータエディタで、Deadline Cloud Render Strategy メニューを使用して送信者の動作を指定できます。

Husk レンダリングと USD ワークフロー

以下のセクションでは、Houdini 送信者の USD エクスポートワークフローの現在の制限と、Husk でエクスポートされた USD シーンをレンダリングするための代替ジョブバンドルの例について説明します。

USD エクスポートワークフローのサポート

Houdini の Deadline Cloud 送信者には、現在、USD エクスポートワークフローのサポートが組み込まれていません。

送信者ノードを使用して、Houdini から USD シーンをエクスポートし、Husk スタンドアロンを呼び出して Houdini エンジンライセンスを消費せずにレンダリングする単一のジョブを作成することはできません。

代替: Husk ジョブバンドルの例

Deadline Cloud は、[Houdini 送信者の外部で USD エクスポートレンダリングワークフローを有効にする Husk ジョブバンドルの例](#)を提供します。サンプルジョブバンドルを使用する前に、Houdini とは別に USD シーンをエクスポートする必要があります。

Husk ジョブバンドルの例:

- Houdini を起動したり、レンダリング中に Houdini エンジンライセンスを消費したりすることなく、Husk と選択したレンダーラーレンダー代理人を使用してレンダリングするための USD シーンを直接送信できます。
- USD ファイルを自動的にイントロスペクションして、ジョブアタッチメントを使用してアタッチする内のファイルの依存関係を見つけます。
- 一般的な Husk 設定と送信を設定するためのシンプルな GUI を提供します。

前提条件

Husk サンプルジョブバンドルを使用する前に、以下が必要です。

- USD 形式にエクスポートされたシーン。
 - Houdini で USD ファイルを書き出す方法については、[SideFX USD ドキュメント](#)を参照してください。
- Deadline Cloud CLI がインストールされ、設定されています。
 - CLI は、送信者インストーラから、または [deadline-cloud 入門ガイド](#)の直後にインストールできます。
- [deadline-cloud-samples リポジトリ](#)の git クローン。
- ワーカーノードで使用可能な、Heridor レンダーの委任。
 - Karma は Houdini に含まれています。他の高機能レンダーの委任を使用する場合は、ワーカーで指定する必要があります。ワーカーノードで使用できるようにするための 1 つのオプションとして、[V-Ray と Redshift の conda パッケージの例については](#)、[deadline-cloud-samples リポジトリ](#)を参照してください。 https://github.com/aws-deadline/deadline-cloud-samples/tree/mainline/conda_recipes/houdini-redshift-2026

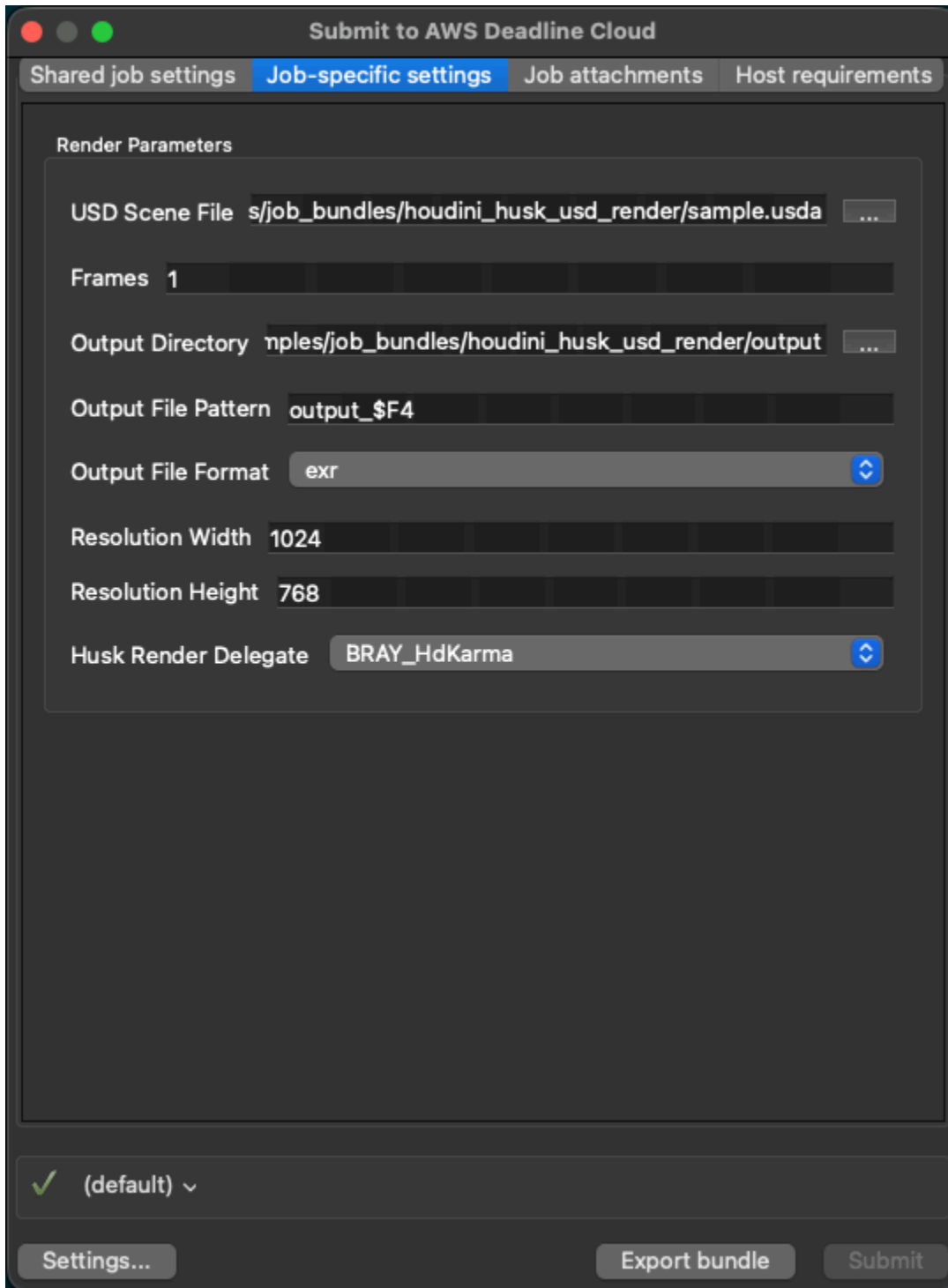
Husk サンプルジョブバンドルの使用

Husk サンプルジョブバンドルを使用するには

1. Deadline Cloud CLI を使用してバンドルを送信します。

```
deadline bundle gui-submit ./deadline-cloud-samples/job_bundles/  
houdini_husk_usd_render
```

2. 送信する USD ファイル、出力設定、フレーム範囲、およびその他の該当する設定を設定します。



その他のリソース

- [deadline-cloud-samples](#) リポジトリ。
- [SideFX Husk](#) ドキュメント。

トラブルシューティング

以下のセクションでは、Houdini の Deadline Cloud 送信者を使用する際に発生する可能性がある一般的なエラーと質問、およびその解決方法について説明します。

レンダリング中に「不完全なアセット定義」エラーが発生するのはなぜですか？

ファームで実行されるこの送信者のジョブでは、ログに次のようなエラーが発生する可能性があります。

```
The following node types are using incomplete asset definitions:  
Driver/deadline_cloud
```

これらのエラーは無視しても安全です。Deadline Cloud 送信者は、Houdini シーンにノードとして存在します。ファームのワーカーがシーンをロードしても、シーンには Deadline Cloud ノードが含まれますが、ワーカーには送信者がインストールされていない可能性があります。ワーカーには Deadline Cloud ノードの実行に必要なファイルがないため、「不完全なアセット定義」エラーがログに記録されます。Deadline Cloud ノード自体はジョブの一部としてレンダリングされないため、これらのエラーは無視できます。

Deadline Cloud 送信者は、Husk を使用した USD エクスポートレンダーワークフローをサポートしていますか？

Houdini 送信者は、現時点では Husk を使用したエクスポートワークフローを直接サポートしていません。送信者を通じて作成されたジョブは、常に `hython` を実行するためのアダプターを実行するため、レンダリングの間中は Houdini エンジンライセンスを使用します。エクスポートされた USD シーンを Husk と高機能レンダーの代理人のみを使用してレンダリングする場合は、[ジョブバンドル](#)の例を使用できます。このアプローチは、レンダリング全体に対して Houdini エンジンライセンスを必要とせず、レンダリングライセンス (Karma など) のみで USD シーンをレンダリングするのに役立ちます。Husk on Deadline Cloud での USD シーンのレンダリングの詳細については、「」を参照してください[Husk レンダリングと USD ワークフロー](#)。

詳細設定

サポートされていないバージョンの使用

Deadline Cloud は、上の表のワークステーションとワーカーソフトウェアのバージョンのみをサポートおよびテストします。送信者を使用する場合、ワーカーはワークステーションと同じバージョン

ンをインストールしようとします。Houdini のワークステーションバージョンが上記のバージョンテーブルに表示されない場合、これは失敗することがあります。

サポートされていないバージョンの Houdini が必要な場合は、次のオプションがあります。

- Houdini からジョブを送信する場合、CondaPackages キューパラメータを上書きして、ワーカーで使用するサポートされているバージョンを指定できます (例: houdini=21.0, houdini-openjd=*)。これは、シーンで使用される機能と、Houdini がワークステーションバージョンのシーンとどのように連携するかに応じて、機能する場合と機能しない場合があります。
- ワーカーにインストールする目的のバージョンのカスタム conda レシピとチャンネルを構築できます。以下でリンクされているサポートされているバージョンの conda レシピを開始点として使用し、目的のバージョンをカスタム conda チャンネルにパッケージ化します。カスタム conda チャンネルの作成の詳細については、[「カスタム conda チャンネルの作成」](#)を参照してください。

Houdini レンダーエンジン

Houdini は、Deadline Cloud と互換性のある複数のレンダーエンジンをサポートしています。

レンダリングエンジン	説明	GPU サポート
Karma CPU	Modern USD ベースのレンダラー (CPU バリエーション)	CPU ベース
Karma XPU	Modern USD ベースのレンダラー (GPU バリエーション)	GPU アクセラレーション
Mantra	従来の Houdini レンダラー	CPU ベース
アーノルド	サードパーティーの Monte Carlo レイトレーサー	GPU/CPU ハイブリッド
V-Ray	サードパーティーのフォトリアリスティックレンダラー	GPU/CPU ハイブリッド
Redshift	GPU アクセラレーションレンダラー	GPU 最適化

これらのレンダリングエンジンは Houdini 統合送信者によって自動的に検出および設定され、使用は自動的にライセンスされます。送信者は、接続されたレンダー出力ノード (ROPs) 間の依存関係ツリーを維持します。

オープンソースリソース

送信者とアダプターはオープンソースであり、GitHub で入手できます。Houdini Conda レシピは、サポートされているバージョンの GitHub で利用できます。

- [GitHub の Houdini 送信者ソースコード](#)
- [GitHub のサンプルシーンとワークフロー](#)
- [GitHub でサポートされているバージョンの Conda レシピ](#)

Deadline Cloud のファイルストレージ

ワーカーは、ジョブの処理に必要な入力ファイルを含むストレージの場所と、出力を保存する場所にアクセスできる必要があります。AWS Deadline Cloud には、次のストレージオプションが用意されています。

- 永続ストレージを使用すると、サービスマネージドフリートワーカーは、ワーカーライフサイクルイベント全体でデータを保持する専用の Amazon Elastic Block Store (Amazon EBS) ボリュームを使用します。アプリケーションキャッシュ、conda パッケージのインストール、ワークスペースは、ワーカーをリサイクルしても保持されるため、コールドスタートの遅延がなくなります。詳細については、「[サービスマネージドフリートの永続的ストレージ](#)」を参照してください。
- ジョブアタッチメントを使用すると、Deadline Cloud はジョブの入出力ファイルをワークステーションと Deadline Cloud ワーカー間で前後に転送します。ファイル転送を有効にするために、Deadline Cloud は Amazon Simple Storage Service (Amazon S3) バケットを使用します AWS アカウント。

Linux ベースのサービスマネージドフリートでジョブアタッチメントを使用する場合、仮想ファイルシステム (VFS) を有効にして、ジョブアタッチメントファイルをマウントし、ジョブの開始時にワーカーに同期するのではなく、必要に応じてそれらにアクセスできます。

- 共有ストレージでは、オペレーティングシステムとのファイル共有を使用してファイルへのアクセスを提供します。

クロスプラットフォーム共有ストレージを使用する場合、ワーカーが 2 つの異なるオペレーティングシステム間のファイルにパスをマッピングできるように、ストレージプロファイルを作成できます。

ホスト設定スクリプトを使用して、LucidLink などのサードパーティーのクラウドストレージソリューションをサービスマネージドフリートと統合することもできます。詳細については、「for M&E ブログ」の「[Deadline Cloud のサービスマネージドフリートスクリプトを使用した LucidLink のセットアップ](#)」を参照してください。AWS

トピック

- [Deadline Cloud のストレージプロファイル](#)
- [Deadline Cloud のジョブアタッチメント](#)

Deadline Cloud のストレージプロファイル

複数のオペレーティングシステムまたは異なるファイルシステムマウントからワークステーションとフリートワーカーホストを使用する場合、ファームにストレージプロファイルを作成して、同じファイルシステムが異なるシステムにマウントされている場所を指定できます。Deadline Cloud は、送信されたワークステーションとは異なるストレージプロファイルでジョブを実行すると、ストレージプロファイルで設定されたディレクトリにあるファイルシステムパスを変換します。

Deadline Cloud ファームでストレージプロファイルを使用すると、次の動作が可能になります。

- キューにジョブを送信すると、ジョブが参照するファイルはワークステーションストレージプロファイルによって分類されます。
 - 共有ファイルシステムの場所にあるファイルは、単独で残ります。
 - ローカルファイルシステムの場所にあるファイルは、ジョブアタッチメント S3 バケットにアップロードすることでジョブにアタッチされます。以前にアップロードされたファイルは再度アップロードされません。
 - ファイルシステムの場所がないファイルもジョブにアタッチされます。ジョブ送信者は、ローカルの Deadline Cloud 設定で既知のパスの下にない限り、これらのファイルパスについて警告します。
- 送信側ワークステーションとは異なるオペレーティングシステムまたはストレージプロファイルを持つフリートワーカーホストでジョブが実行されている場合、ジョブで使用されるファイルパスは送信側ストレージプロファイルからフリートワーカーホストのストレージプロファイルにマッピングされます。
- ジョブ出力をダウンロードすると、別のオペレーティングシステムまたはストレージプロファイルに送信されたジョブは、送信先のストレージプロファイルからローカルワークステーションのストレージプロファイルにパスがマッピングされます。

詳細については、AWS Deadline Cloud デベロッパーガイドの [「ストレージプロファイルとパスマッピング」](#) を参照してください。

ストレージプロファイルを作成するには

1. [Deadline Cloud コンソール](#)を開きます。
2. 開始するには、「Deadline Cloud ダッシュボードに移動」を選択します。
3. ファームを選択し、ストレージプロファイルタブを選択します。
4. ストレージプロファイルの作成 を選択します。
5. ドロップダウンからオペレーティングシステムを選択します。

6. ストレージプロファイル名を入力します。名前は、ワークステーションのストレージプロファイルを選択する方法です。たとえば、Windows-Workstation や Windows-OnPremFleet などの名前を使用すると、後で簡単に識別できます。
7. ワークステーションとフリートワーカーホストの両方にマウントされる共有ファイルシステムごとに、共有タイプのファイルシステムの場所を作成します。
 1. プロジェクトデータを含む共有ファイルシステムのプロジェクトや、使用するツールを含む共有ファイルシステムのツールなど、マウントを識別する名前を入力します。
 2. ストレージプロファイルのオペレーティングシステムで、選択した共有ファイルシステムのマウント場所を入力します。
8. ワークステーション専用の共有ファイルシステムごとに、ローカルタイプのファイルシステムの場所を作成します。たとえば、フリートがオンになって AWS いて、ジョブアタッチメントでデータ転送を処理する場合です。また、各ワークステーションにローカルなディレクトリにこの種のファイルシステムの場所を作成して、異なるオペレーティングシステムで同等のパスを指定することもできます。
 1. プロジェクトデータを含む共有ファイルシステムのプロジェクトや、使用するツールを含む共有ファイルシステムのツールなど、マウントを識別する名前を入力します。
 2. ストレージプロファイルのオペレーティングシステムで、選択したファイルシステムの場所を入力します。
9. (オプション) 別のファイルシステムの場所を追加するには、新しい必要なファイルシステムの場所を追加を選択し、必要なデータを入力します。
10. 必要なファイルシステムの場所をすべて追加したら、作成を選択します。

ストレージプロファイルを使用するためにセットアップするには

1. このストレージプロファイルを使用するキューに移動し、許可されたストレージプロファイルタブを選択します。
2. ストレージプロファイルの設定 を選択します。
3. ドロップダウンリストから、作成したストレージプロファイルを選択します。
4. 必須ファイルシステムの場所リストで、関連付けられたフリートの任意のストレージプロファイルで が使用可能であることを確認するファイルシステムの場所名を選択します。
5. (オプション) フリートのストレージプロファイルを作成した場合は、フリートに移動し、設定タブを選択します。

- a. Storage profiles セクションで、Configure storage profile を選択します。
- b. ストレージプロファイルを選択し、変更の保存を選択します。

ワークステーションでストレージプロファイルを設定するには

キューにジョブを送信する各ワークステーションで、設定ダイアログを使用してデフォルトのストレージプロファイルを選択します。

1. Deadline Cloud 設定ダイアログを開くには、次のいずれかの手順を実行します。
 - a. Deadline Cloud 送信者の設定ボタンを選択します。

または
 - b. `deadline config gui` CLI コマンドを実行します。
2. デフォルトのファームとキューを設定したら、ドロップダウンリストからデフォルトのストレージプロファイルを選択します。

共有ファイルシステムのストレージプロファイル

[サービスマネージドフリートで VPC リソースエンドポイントを使用するか](#)、またはオンプレミスでカスターマネージドフリートのホストを設定することで、共有ファイルシステムをマウントするように Deadline Cloud AWS フリートを設定できます。ワークステーションにフリートと同じ共有ファイルシステムがマウントされている場合、ストレージプロファイルに共有タイプのファイルシステムの場所を作成して、各共有ファイルシステムがローカルパスとして表示される場所を設定できます。

たとえば、プロジェクト用に 1 つの共有ファイルシステムがあり、ツール用に別の共有ファイルシステムがあるとします。ワークステーションとフリートには、3 つのオペレーティングシステム Windows、macOS、および Linux が含まれています。次の値を使用して、オペレーティングシステムごとに 1 つのストレージプロファイルを作成できます。

- ストレージプロファイル名: Linux-Host、オペレーティングシステムファミリー: Linux。
 - ファイルシステムの場所名: プロジェクト、パス: `/mnt/projects`、タイプ: 共有。
 - ファイルシステムの場所名: ツール、パス: `/mnt/projects`、タイプ: 共有。
- ストレージプロファイル名: Windows-Host、オペレーティングシステムファミリー: Windows。
 - ファイルシステムの場所名: プロジェクト、パス: `X:\projects`、タイプ: 共有。
 - ファイルシステムの場所名: ツール、パス: `Z:`、タイプ: 共有。

- ストレージプロファイル名: MacOS-Host、オペレーティングシステムファミリー: MacOS。
 - ファイルシステムの場所名: プロジェクト、パス: /ボリューム/プロジェクト、タイプ: 共有。
 - ファイルシステムの場所名: Tools、path: /Volumes/Tools、type: Shared。

パス X:\Projects\ProjectA\Textures\texture.jpg Windows を使用するジョブを から送信すると、Deadline Cloud は Windows-Host ストレージプロファイル ID を含むフィールドをジョブに追加します。

ジョブが Linux フリートワーカーホストで実行される場合、Deadline Cloud は対応するファイルシステムの場所名に基づいてジョブの 2 つのパスマッピングルールを作成します。X:\Projects -> /mnt/projects、Z: -> /mnt/tools。ジョブは、これらのルールを適用して、Linux ホストがそれらを表示する元のパスを解決します。

ジョブアタッチメントがキューにも設定されている場合、共有タイプのファイルシステムの場所がないパスはジョブにアタッチされ、ジョブアタッチメント S3 バケットにアップロードされます。これにより、常に共有ファイルシステムにコピーする必要なく、データファイルをジョブにアタッチできます。たとえば、送信するジョブバンドルで定義された補助ファイルを指定します。

ジョブアタッチメントのストレージプロファイル

Deadline Cloud キューを設定して、ジョブによって参照されるアセットデータを との間で転送するためのジョブアタッチメントを使用できます AWS。ワークステーションが同じ共有ファイルシステムをマウントしてもフリートがマウントしない場合は、ストレージプロファイルにローカルタイプのファイルシステムの場所を作成できます。この設定では、ファイルのアップロード元とダウンロード元、およびオペレーティングシステム間のパスをマッピングする方法を設定できます。

たとえば、プロジェクト用に 1 つの共有ファイルシステムがあり、ツール用に別の共有ファイルシステムがあるとします。ワークステーションとフリートには、3 つのオペレーティングシステム Windows、macOS、および が含まれています Linux。ファイルシステムがファームと共有されない点を除いて、共有ファイルシステムのストレージプロファイルのトピックと同じです。これらは、ワークステーションを含むローカルエリアネットワーク用です。次の値を使用して、オペレーティングシステムごとに 1 つのストレージプロファイルを作成できます。

- ストレージプロファイル名: Linux-Host、オペレーティングシステムファミリー: Linux。
 - ファイルシステムの場所名: プロジェクト、パス: /mnt/projects、タイプ: Local。
 - ファイルシステムの場所名: ツール、パス: /mnt/projects、タイプ: Local。
- ストレージプロファイル名: Windows-Host、オペレーティングシステムファミリー: Windows。

- ファイルシステムの場所名: プロジェクト、パス: X:\projects、タイプ: Local。
- ファイルシステムの場所名: ツール、パス: Z:、タイプ: Local。
- ストレージプロファイル名: MacOS-Host、オペレーティングシステムファミリー: MacOS。
 - ファイルシステムの場所名: プロジェクト、パス: /Volumes/Projects、タイプ: Local。
 - ファイルシステムの場所名: Tools、パス: /Volumes/Tools、タイプ: Local。

パス X:\Projects\ProjectA\Textures\texture.jpg Windowsを使用するジョブを から送信すると、Deadline Cloud は Windows-Host ストレージプロファイル ID を含むフィールドをジョブに追加し、まだアップロードされていない場合はジョブアタッチメント S3 バケットにファイルをアップロードします。

ジョブがLinuxフリートワーカーホストで実行されている場合、Deadline Cloud はテクスチャファイルをローカル一時ディレクトリで使用できるようにし、テクスチャを含むディレクトリの1つから一時ディレクトリへのパスマッピングルールを作成します。たとえば、X:\Projects\ProjectA -> /sessions/session-123/projects の場合、X:\Projects\ProjectA\Textures\texture.jpg は /sessions/session-123/projects/Textures/texture.jpg にマッピングされます。ジョブのタスクが完了すると、ジョブで指定されたディレクトリから出力が収集されます。/sessions/session-123/projects/Output/frame0032.png が出力ファイルであるとします。この出力は、ジョブを送信するワークステーションのストレージプロファイルと一致する X:\Projects\ProjectA\Output\frame0032.jpg としてジョブに記録されます。

macOS ワークステーションにジョブ出力をダウンロードすると、Deadline Cloud はWindowsワークステーションからパスマッピングルールを作成します: X:\Projects -> /Volumes/Projects、Z: -> /Volumes/Tools。ルールをすべての出力パスに適用し、サンプル出力ファイルを /Volumes/Projects/ProjectA/Output/frame0032.jpg にダウンロードします。

ジョブの出力ファイルパスがストレージプロファイルファイルシステムの場所のいずれにも含まれていない場合、Deadline Cloud は、ストレージプロファイルが送信先のワークステーションと異なる場合、ダウンロードするパスを決定できません。ダウンロードに使用するコマンドに応じて、そのファイルはスキップされるか、ダウンロードディレクトリを手動で選択する必要があります。

Deadline Cloud のジョブアタッチメント

ジョブアタッチメントを使用すると、ワークステーションと AWS Deadline Cloud 間でファイルを前後に転送できます。ジョブアタッチメントを使用すると、ファイル用に Amazon S3 バケットを手動で設定する必要はありません。代わりに、Deadline Cloud コンソールでキューを作成するときに、ジョブアタッチメントのバケットを選択します。

Deadline Cloud にジョブを初めて送信すると、ジョブのすべてのファイルが Deadline Cloud に転送されます。後続の送信では、変更されたファイルのみが転送されるため、時間と帯域幅の両方を節約できます。

処理が完了したら、ジョブの詳細ページから、または Deadline Cloud CLI `deadline job download-output` コマンドを使用して結果をダウンロードできます。

複数のキューに同じ S3 バケットを使用できます。バケット内の添付ファイルを整理するには、キューごとに異なるルートプレフィックスを設定します。

コンソールでキューを作成するときは、既存の AWS Identity and Access Management (IAM) ロールを選択するか、コンソールに新しいロールを作成させることができます。コンソールがロールを作成すると、キューに指定されたバケットにアクセスするためのアクセス許可が設定されます。既存のロールを選択する場合は、S3 バケットにアクセスするためのアクセス許可をロールに付与する必要があります。

ジョブアタッチメント S3 バケットの暗号化

ジョブアタッチメントファイルは、デフォルトで S3 バケットで暗号化されます。この暗号化は、不正アクセスから情報を保護するのに役立ちます。Deadline Cloud が提供するキーでファイルを暗号化するために何もする必要はありません。詳細については、「[Amazon S3 ユーザーガイド](#)」の「[Amazon S3 ですべての新しいオブジェクトが自動的に暗号化](#)」を参照してください。

独自のカスタマーマネージド AWS Key Management Service キーを使用して、ジョブアタッチメントを含む S3 バケットを暗号化できます。そのためには、バケットに関連付けられたキューの IAM ロールを変更して、へのアクセスを許可する必要があります AWS KMS key。

キューロールの IAM ポリシーエディタを開くには

1. にサインイン AWS マネジメントコンソール し、Deadline Cloud [コンソール](#)を開きます。メインページの開始方法セクションで、ファームの表示を選択します。
2. ファームのリストから、変更するキューを含むファームを選択します。
3. キューのリストから、変更するキューを選択します。
4. キューの詳細セクションで、サービスロールを選択して、サービスロールの IAM コンソールを開きます。

次に、次の手順を実行します。

のアクセス許可でロールポリシーを更新するには AWS KMS

1. アクセス許可ポリシーのリストから、ロールのポリシーを選択します。
2. このポリシーで定義されているアクセス許可セクションで、**編集** を選択します。
3. **[新しいステートメントを追加]** を選択します。
4. 次のポリシーをコピーしてエディタに貼り付けます。*Region*、*accountID*、を独自の値*keyID*に変更します。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:GenerateDataKey"
  ],
  "Resource": [
    "arn:aws:kms:us-east-1:111122223333:key/keyID"
  ]
}
```

5. **[次へ]** を選択します。
6. ポリシーの変更を確認し、問題がなければ変更を保存を選択します。

ジョブアタッチメントバケットを置き換える

現在のジョブアタッチメントバケットを別のジョブアタッチメントバケットに置き換えることができます。キューの詳細のジョブアタッチメントタブの下にボタンがあります。これを使用して、ジョブアタッチメントバケットを変更するか、同じバケット内のルートフォルダを置き換えてジョブアタッチメントをアップロードできます。

ジョブアタッチメント設定にアクセスするには

1. キューの詳細に移動し、ジョブアタッチメントタブを見つけます。
2. ジョブアタッチメントタブには、次の2つのオプションがあります。
 - a. ジョブアタッチメントバケットを変更するには、次の手順を実行します。
 - i. 新しい S3 バケットを選択します。

- ii. キューのサービスロールポリシーを更新して、新しいバケットへのアクセスを許可します。

OR

- b. 以下を実行して、既存のバケット内のルートフォルダを変更します。
 - i. ルートフォルダ名を変更します。
 - ii. キューサービスロールのリソース ARN を更新します。

サービスロールを更新するには

1. ファーム > キュー > キューサービスロールに移動します。
2. [JSON で編集] を選択します。
3. リソース ARN を見つけます (デフォルトのルートフォルダは DeadlineCloud です) 。

```
"arn:aws:s3:::<your-job-attachments-bucket-name>/DeadlineCloud/*"  
]
```

4. 新しいバケットまたはフォルダで ARN を更新します。

```
"arn:aws:s3:::<your-job-attachments-NEW-bucket-name>/NEW-ROOT-FOLDER-NAME/*"  
]
```

5. これらの変更を行った後にアクセス許可を確認して、適切なアクセスを確保します。

S3 バケットでのジョブアタッチメントの管理

Deadline Cloud は、ジョブに必要なジョブアタッチメントファイルを S3 バケットに保存します。これらのファイルは時間の経過とともに蓄積されるため、Amazon S3 のコストが増加します。コストを削減するために、S3 バケットに S3 ライフサイクル設定を適用できます。この設定では、バケット内のファイルを自動的に削除できます。S3 バケットはアカウントにあるため、いつでも S3 ライフサイクル設定を変更または削除できます。詳細については、「[Amazon S3 ユーザーガイド](#)」の「[S3 ライフサイクル設定の例](#)」を参照してください。Amazon S3

より詳細な S3 バケット管理ソリューションでは、S3 バケット内のオブジェクトが最後にアクセスされた時刻に基づいて AWS アカウント 期限切れになるようにを設定できます。詳細について

は、アーキテクチャブログの「[最終アクセス日に基づく Amazon S3 オブジェクトの有効期限切れ AWS](#)」を参照してください。

Deadline Cloud 仮想ファイルシステム

Deadline Cloud AWS でのジョブアタッチメントの仮想ファイルシステムサポートにより、ワーカーのクライアントソフトウェアが Amazon Simple Storage Service と直接通信できるようになります。ワーカーは、処理前にすべてのファイルをダウンロードするのではなく、必要な場合にのみファイルをロードできます。ファイルはローカルに保存されます。このアプローチにより、複数回使用されるアセットのダウンロードを回避できます。ジョブが完了すると、すべてのファイルが削除されます。

- 仮想ファイルシステムは、特定のジョブプロファイルのパフォーマンスを大幅に向上させます。一般的に、ワーカーのフリートが大きいファイルの合計のサブセットが小さいほど、最も利点があります。ワーカー数が少ない少数のファイルでは、ほぼ同等の処理時間があります。
- 仮想ファイルシステムのサポートは、サービスマネージドフリートのLinuxワーカーのみが使用できます。
- Deadline Cloud 仮想ファイルシステムは、以下のオペレーションをサポートしていますが、POSIX に準拠していません。
 - ファイル
create、delete、open、close、read、write、append、truncate、rename、move、copy および fallop
 - ディレクトリ create、delete、rename、movecopy、および stat
- 仮想ファイルシステムは、タスクが大規模なデータセットの一部にのみアクセスする場合のデータ転送を減らし、パフォーマンスを向上させるように設計されており、すべてのワークロードに最適化されているわけではありません。本番稼働用ジョブを実行する前に、ワークロードをテストする必要があります。

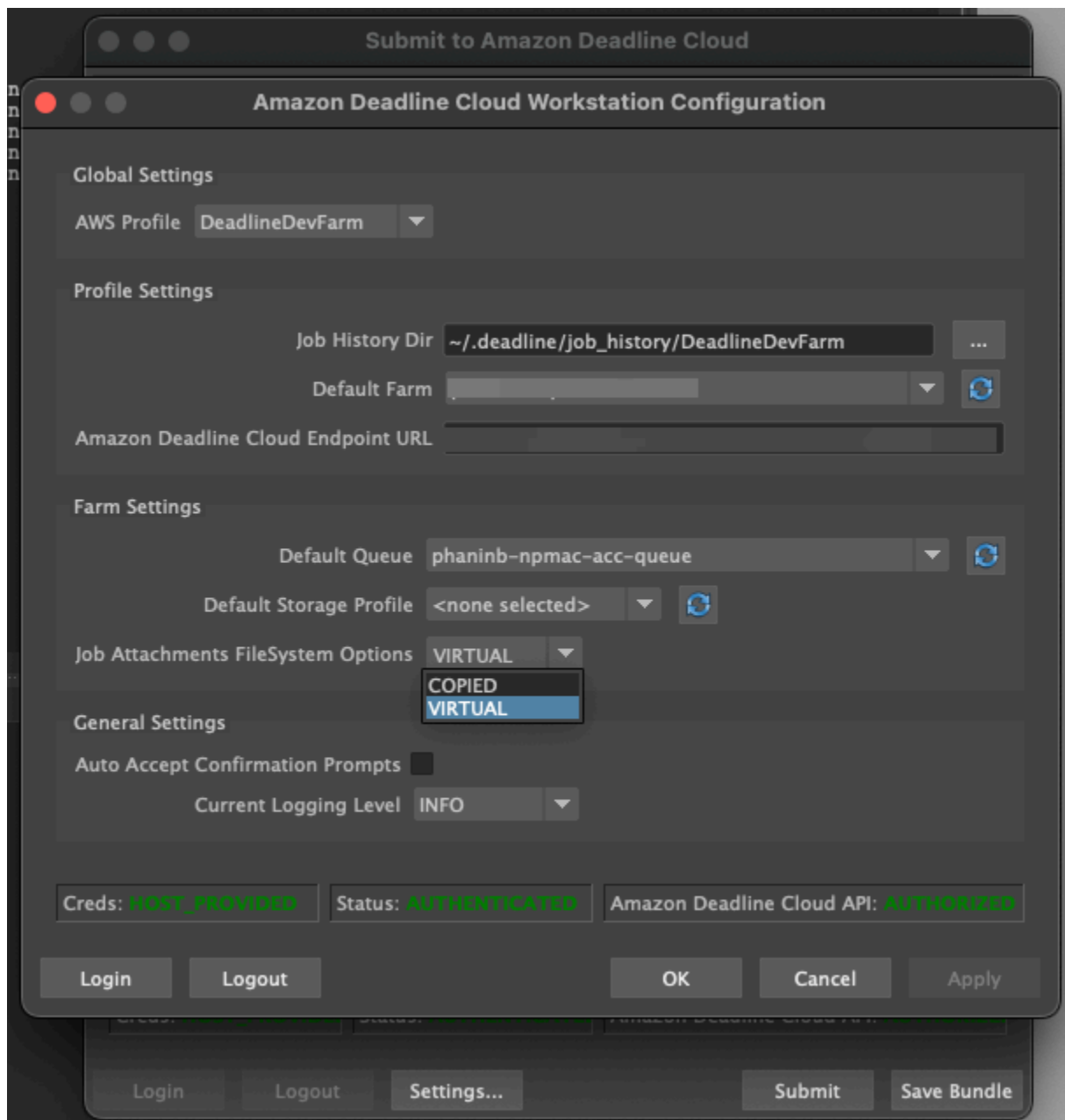
VFS サポートを有効にする

仮想ファイルシステムサポート (VFS) はジョブごとに有効になっています。このような場合、ジョブはデフォルトのジョブアタッチメントフレームワークにフォールバックします。

- ワーカーインスタンスプロファイルは、仮想ファイルシステムをサポートしていません。
- 問題により、仮想ファイルシステムプロセスを起動できません。
- 仮想ファイルシステムはマウントできません。

送信者を使用して仮想ファイルシステムのサポートを有効にするには

1. ジョブを送信するときは、設定ボタンを選択して AWS Deadline Cloud ワークステーションの設定パネルを開きます。
2. ジョブアタッチメントのファイルシステムオプションドロップダウンから、VIRTUAL を選択します。



3. 変更を保存するには、OK を選択します。

を使用して仮想ファイルシステムのサポートを有効にするには AWS CLI

- 保存したジョブを送信するときは、次のコマンドを使用します。

```
deadline bundle submit-job --job-attachments-file-system VIRTUAL
```

仮想ファイルシステムが特定のジョブに対して正常に起動されたことを確認するには、Amazon CloudWatch Logs でログを確認します。次のメッセージを探します。

```
Using mount_point mount_point  
Launching vfs with command command  
Launched vfs as pid PID number
```

ログに次のメッセージが含まれている場合、仮想ファイルシステムのサポートは無効になります。

```
Virtual File System not found, falling back to COPIED for JobAttachmentsFileSystem.
```

仮想ファイルシステムサポートのトラブルシューティング

Deadline Cloud モニターを使用して、仮想ファイルシステムのログを表示できます。手順については、「[Deadline Cloud でセッションログとワーカーログを表示する](#)」を参照してください。

仮想ファイルシステムログは、ワーカーエージェントの出力と共有されているキューに関連付けられている CloudWatch Logs グループにも送信されます。

自動ダウンロード

Deadline CLI は、同じコマンドが最後に実行されてから完了したキュー内のすべてのタスクの出力をダウンロードするコマンドを提供します。これは、繰り返し実行する cron ジョブまたはスケジュールされたタスクとして設定できます。この設定では、出力の自動ダウンロードが継続的に設定されます。

自動ダウンロードを設定する前に、[ジョブアタッチメントのストレージプロファイル](#)の手順に従って、アップロードおよびダウンロードするアセットデータのすべてのパスを設定します。ジョブがストレージプロファイルにない出力パスを使用する場合、自動ダウンロードはその出力のダウンロードをスキップし、警告メッセージを出力して、ダウンロードしなかったファイルを要約します。同様に、ストレージプロファイルなしでジョブが送信された場合、自動ダウンロードはそのジョブをスキップし、警告メッセージを出力します。デフォルトでは、Deadline Cloud 送信者は、正しい設定を保証するために、ストレージプロファイルの外部にあるパスの警告メッセージを表示します。

AWS 認証情報の設定

自動ダウンロードでは、Deadline CLI を使用してジョブ出力を継続的にダウンロードします。これらのダウンロードを認証するには、長期的な IAM 認証情報が必要です。Deadline Cloud Monitor の認証情報の有効期限が切れるため、この目的で使用することはできません。

長期認証情報を設定するには、次の手順に従います。

Important

次の警告に注意してください。

- アカウントのルート認証情報を使用して AWS リソースにアクセスしないでください。これらの認証情報は無制限のアカウントアクセスを提供し、取り消すのが困難です。
- アプリケーションファイルにリテラルアクセスキーや認証情報を配置しないでください。これを行うと、パブリックリポジトリにプロジェクトをアップロードするなど、誤って認証情報が公開されるリスクが発生します。
- プロジェクト領域に認証情報を含むファイルを含めないでください。
- アクセスキーを保護します。[アカウント識別子を確認する](#)ためであっても、アクセスキーを認可されていない当事者に提供しないでください。提供すると、第三者がアカウントへの永続的なアクセスを取得する場合があります。
- 共有 AWS 認証情報ファイルに保存されている認証情報はすべてプレーンテキストで保存されることに注意してください。

詳細については、[AWS 全般のリファレンスの AWS 「アクセスキーを管理するためのベストプラクティス」](#)を参照してください。

IAM ユーザーの作成

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで [ユーザー]、[ユーザーの作成] の順に選択します。
3. ユーザー に名前を付けます **deadline-output-downloader**。へのユーザーアクセスを許可する AWS マネジメントコンソールのチェックボックスをオフにし、次へを選択します。
4. [ポリシーを直接アタッチ] を選択します。
5. ポリシーの作成 を選択して、最低限必要なアクセス許可を持つカスタムポリシーを作成します。

6. JSON エディタで、次のアクセス許可を指定します。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DeadlineCloudOutputDownload",
            "Effect": "Allow",
            "Action": [
                "deadline:AssumeQueueRoleForUser",
                "deadline:ListQueueEnvironments",
                "deadline:ListSessions",
                "deadline:ListSessionActions",
                "deadline:SearchJobs",
                "deadline:GetJob",
                "deadline:GetQueue",
                "deadline:GetStorageProfileForQueue"
            ],
            "Resource": "*"
        }
    ]
}
```

7. ポリシーに名前を付け **DeadlineCloudOutputDownloadPolicy**、ポリシーの作成を選択します。
8. ユーザー作成ページに戻り、ポリシーリストを更新して、先ほど作成した **DeadlineCloudOutputDownloadPolicy** を選択し、次へを選択します。
9. ユーザーの詳細を確認し、ユーザーの作成を選択します。

アクセスキーの作成

1. ユーザーの詳細ページから、セキュリティ認証情報タブを選択します。[Access keys (アクセスキー)] セクションで、[Create access key (アクセスキーを作成)] を選択します。
2. Other に キーを使用するように指定し、Next を選択し、Create access key を選択します。
3. アクセスキーの取得ページで、表示を選択してユーザーのシークレットアクセスキーの値を表示します。認証情報をコピーするか、csv ファイルをダウンロードできます。

ユーザーアクセスキーを保存する

- ユーザーアクセスキーをシステムの AWS 認証情報ファイルに保存します。
 - ではLinux、ファイルは にあります。 ~/.aws/credentials
 - ではWindows、ファイルは にあります。 %USERPROFILE\.aws\credentials

次のキーを置き換えます。

```
[deadline-downloader]
aws_access_key_id=ACCESS_KEY_ID
aws_secret_access_key=SECRET_ACCESS_KEY
region=YOUR_AWS_REGION
```

Important

この IAM ユーザーが不要になった場合は、[AWS セキュリティのベストプラクティス](#)に合わせて削除することをお勧めします。にアクセスする[AWS IAM アイデンティティセンター](#)ときは、を通じて一時的な認証情報を使用することを人間のユーザーに要求することをお勧めします AWS。

前提条件

自動ダウンロード用の cron ジョブまたはスケジュールされたタスクを作成する前に、次の手順を実行します。

1. まだインストールしていない場合は、[Python](#) をインストールします。
2. 以下を実行して Deadline CLI をインストールします。

```
python -m pip install deadline
```

3. 次のコマンドを使用して、Deadline CLI のバージョンが 0.52.1 以降であることを確認します。

```
$ deadline --version
deadline, version 0.52.1
```

出力ダウンロードコマンドをテストする

環境でコマンドが動作することを確認するには

1. Deadline へのパスを取得する

Linux and macOS

```
$ which deadline
```

Windows

```
C:\> where deadline
```

PowerShell

```
PS C:\> Get-Command deadline
```

2. sync-output コマンドを実行してブートストラップします。

```
/path/to/deadline queue sync-output \  
--profile deadline-downloader \  
--farm-id YOUR_FARM_ID \  
--queue-id YOUR_QUEUE_ID \  
--storage-profile-id YOUR_PROFILE_ID \  
--checkpoint-dir /path/to/checkpoint/directory \  

```

3. ダウンロードマシンが送信マシンと同じ場合にのみ、このステップを実行する必要があります。--storage-profile-id YOUR_PROFILE_ID \ 上記のを に置き換えます--ignore-storage-profiles。

4. テストジョブを送信します。

a. GitHub から .zip ファイルをダウンロードします。

i. [deadline-cloud-samples GitHub リポジトリ](#)を開きます。

ii. Code を選択し、ドロップダウンメニューから Download ZIP を選択します。

iii. ダウンロードしたアーカイブをローカルディレクトリに解凍します。

b. 実行

```
cd /path/to/unzipped/deadline-cloud-samples-mainline/job_bundles/  
job_attachments_devguide_output
```

c. 実行

```
deadline bundle submit .
```

- デフォルトの期限設定がない場合は、コマンドラインで以下を指定する必要があります。

```
--farm-id YOUR-FARM-ID --queue-id YOUR-QUEUE-ID
```

d. ジョブが完了するまで待ってから、次のステップに進みます。

5. sync-output コマンドを再度実行します。

```
/path/to/deadline queue sync-output \  
--profile deadline-downloader \  
--farm-id YOUR_FARM_ID \  
--queue-id YOUR_QUEUE_ID \  
--storage-profile-id YOUR_PROFILE_ID \  
--checkpoint-dir /path/to/checkpoint/directory
```

6. 以下について確認します。

- テストジョブの出力は、送信先ディレクトリに表示されます。
- チェックポイントファイルは、指定されたチェックポイントディレクトリに作成されます。

スケジュールされたダウンロードを設定する

オペレーティングシステムのタブを選択すると、5 分ごとに自動ダウンロードを設定する方法がわかります。

Linux

1. Deadline CLI のインストールを確認する

期限実行ファイルへの正確なパスを取得します。

```
$ which deadline
```

このパス (例: /opt/homebrew/bin/deadline) を plist ファイルで使用することに注意してください。

2. チェックポイントディレクトリの作成

チェックポイントファイルを保存するディレクトリを作成します。ユーザーが コマンドを実行するための適切なアクセス許可があることを確認します。

```
$ mkdir -p /path/to/checkpoint/directory
```

3. ログディレクトリの作成

cron ジョブログのディレクトリを作成します。

```
$ mkdir -p /path/to/logs
```

<https://www.redhat.com/en/blog/setting-logrotate> を使用してログファイルのログローテーションを設定することを検討してください。

4. 現在の Crontab を確認する

現在の crontab を表示して、既存のジョブを表示します。

```
$ crontab -l
```

5. Crontab の編集

crontab ファイルを開いて編集します。

```
$ crontab -e
```

初めての場合は、エディタ (nano、vim など) を選択するように求められる場合があります。

6. Cron ジョブエントリの追加

次の行を追加して 5 分ごとにジョブを実行します (パスをステップ 1 と 2 の実際の値に置き換えます)。

```
*/5 * * * * /path/to/deadline queue sync-output --profile deadline-downloader  
--farm-id YOUR_FARM_ID --queue-id YOUR_QUEUE_ID --storage-profile-id
```

```
YOUR_PROFILE_ID --checkpoint-dir /path/to/checkpoint/directory >> /path/to/
logs/deadline_sync.log 2>&1
```

7. Cron ジョブのインストールを確認する

エディタを保存して終了したら、cron ジョブが追加されていることを確認します。

```
$ crontab -l
```

新しいジョブが一覧表示されます。

8. Cron サービスのステータスを確認する

cron サービスが実行されていることを確認します。

```
# For systemd systems (most modern Linux distributions)
$ sudo systemctl status cron
# or
$ sudo systemctl status crond

# For older systems
$ sudo service cron status
```

実行されていない場合は、起動します。

```
$ sudo systemctl start cron
$ sudo systemctl enable cron # Enable auto-start on boot
```

macOS

1. Deadline CLI のインストールを確認する

期限実行ファイルへの正確なパスを取得します。

```
$ which deadline
```

このパス (例: /opt/homebrew/bin/deadline) を plist ファイルで使用することに注意してください。

2. チェックポイントディレクトリとログディレクトリを作成する

チェックポイントファイルを保存するディレクトリを作成します。

```
$ mkdir -p /path/to/checkpoint/directory
$ mkdir -p /path/to/logs
```

<https://formulae.brew.sh/formula/logrotate> を使用してログファイルのログローテーションを設定することを検討してください。

3. リストファイルを作成する

次の内容~/Library/LaunchAgents/com.user.deadlinesync.plistで に設定ファイルを作成します (をステップ 1 の実際のパス/path/to/deadlineに置き換えます)。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key>
  <string>com.user.deadlinesync</string>
  <key>ProgramArguments</key>
  <array>
    <string>/path/to/deadline</string>
    <string>queue</string>
    <string>sync-output</string>
    <string>--profile</string>
    <string>deadline-downloader</string>
    <string>--farm-id</string>
    <string>YOUR_FARM_ID</string>
    <string>--queue-id</string>
    <string>YOUR_QUEUE_ID</string>
    <string>--storage-profile-id</string>
    <string>YOUR_STORAGE_PROFILE_ID</string>
    <string>--checkpoint-dir</string>
    <string>/path/to/checkpoint/dir</string>
  </array>
  <key>RunAtLoad</key>
  <true/>
  <key>UserName</key>
  <string>YOUR_USER_NAME</string>
  <key>StandardOutPath</key>
  <string>/path/to/logs/deadline_sync.log</string>
```

```
<key>StartInterval</key>
<integer>300</integer>
</dict>
</plist>
```

ダウンロードするマシンが送信マシンと同じ`--ignore-storage-profiles`場合は、`--storage-profile-id` *YOUR_PROFILE_ID*上記のを に置き換えます。

4. リストファイルを検証する

plist ファイルの XML 構文を検証します。

```
$ plutil -lint ~/Library/LaunchAgents/com.user.deadlinesync.plist
```

ファイルが有効な場合、「OK」が返されます。

5. 既存の起動エージェントまたは起動デーモンを確認する

起動エージェントが既にロードされているかどうかを確認します。

```
$ launchctl list | grep deadlinesync
OR
$ sudo launchctl list | grep deadlinesync
```

存在する場合は、まずアンロードします。

```
$ launchctl bootout gui/${id -u}/com.user.deadlinesync
OR
$ sudo launchctl bootout system/com.user.deadlinesync
```

6. 作成とブートストラップ

ユーザーがログインしている間にこのタスクを実行するには、LaunchAgent として実行します。マシンが実行されるたびにユーザーがログインせずにこのタスクを実行するには、LaunchDaemon として実行します。

a. LaunchAgent として実行するには:

- i. で作成された設定を使用する `~/Library/LaunchAgents/com.user.deadlinesync.plist`
- ii. 次に、ブートストラップコマンドを使用して設定をロードします。

```
$ launchctl bootstrap gui/$(id -u) ~/Library/LaunchAgents/  
com.user.deadlinesync.plist
```

b. LaunchDaemon として実行するには:

i. 以下を実行して、Plist ファイルを移動し、アクセス許可を変更します。

```
$ sudo mv ~/Library/LaunchAgents/com.user.deadlinesync.plist /Library/  
LaunchDaemons/  
$ sudo chown root:wheel /Library/LaunchDaemons/  
com.user.deadlinesync.plist  
$ sudo chmod 644 /Library/LaunchDaemons/com.user.deadlinesync.plist
```

ii. 最新のブートストラップコマンドを使用して起動エージェントをロードします。

```
$ sudo launchctl bootstrap system /Library/LaunchDaemons/  
com.user.deadlinesync.plist
```

7. ステータスの確認

LaunchAgent をブートストラップした場合は、以下を実行してロードされていることを確認します。

```
$ launchctl list | grep deadlinesync
```

LaunchDaemon をブートストラップした場合は、以下を実行してロードされていることを確認します。

```
$ sudo launchctl list | grep deadlinesync
```

出力は次のようになります。

```
SOME_PID_NUMBER 0 com.user.deadlinesync
```

詳細なステータス情報については、以下を参照してください。

```
$ launchctl print gui/$(id -u)/com.user.deadlinesync
```

現在の状態、プログラム引数、環境変数、実行間隔、実行履歴が表示されます。

Windows

Note

これらの手順を使用して作成されたスケジュールされたタスクは、ユーザーがログインしている場合にのみ機能します。

ユーザーログインを必要とせずにシステム起動時にセットアップするには、公式 [Windows ドキュメント](#) を参照してください。

以下のすべてのステップで、コマンドプロンプトを使用します。管理者として実行します。

1. Deadline CLI のインストールを確認する

期限の実行可能ファイルを見つけます。

```
C:\> where deadline
```

タスクで使用するフルパス (例: C:\Program Files\Amazon\DeadlineCloud\deadline.exe) を書き留めます。

2. チェックポイントディレクトリの作成

チェックポイントファイルを保存するディレクトリを作成します。

```
C:\> mkdir "path\to\checkpoint\directory"
```

3. ログディレクトリの作成

タスクログのディレクトリを作成します。

```
C:\> mkdir "path\to\logs"
```

4. バッチファイルラッパーの作成

次の内容のバッチファイルを作成します。

```
C:\> notepad C:\path\to\deadline_sync.bat
```

```
YOUR_PATH_TO_DEADLINE.EXE queue sync-output --profile deadline-downloader  
--farm-id YOUR_FARM_ID --queue-id YOUR_QUEUE_ID --storage-profile-
```

```
id YOUR_PROFILE_ID --checkpoint-dir path\to\checkpoint\checkpoints > path\to\logs\deadline.log 2>&1
```

5. バッチファイルのテスト

バッチファイルを手動でテストします。

```
C:\> .\path\to\deadline_sync.bat
```

ログファイルが作成されたことを確認します。

```
C:\> notepad path\to\logs\deadline_sync.log
```

6. タスクスケジューラサービスの確認

Task Scheduler サービスが実行されていることを確認します。

```
C:\> sc query "Schedule"
```

サービスが存在しない場合は、代替名を試してください。

```
C:\> sc query "TaskScheduler"  
C:\> sc query "Task Scheduler"
```

実行されていない場合は、起動します。

```
C:\> sc start "Schedule"
```

7. スケジュールされたタスクの作成

5分ごとに実行するタスクを作成します。

```
C:\> schtasks /create /tn "DeadlineOutputSync" /tr "C:\path\to\deadline_sync.bat" /sc minute /mo 5
```

コマンドの内訳:

- /tn - タスク名
- /tr - 実行するタスク (バッチファイル)

- /sc minute /mo 5 - スケジュール: 5 分ごと

8. タスク作成の検証

タスクが正常に作成されたことを確認します。

```
schtasks /query /tn "DeadlineOutputSync" /v /fo LIST
```

以下を探します。

- 実行するタスク: バッチファイルパスを表示します
- 次の実行時間: 5 分以内に時間を表示する必要があります

9. テストタスクの実行

タスクを手動で実行してテストします。

```
schtasks /run /tn "DeadlineOutputSync"
```

タスクのステータスを確認します。

```
schtasks /query /tn "DeadlineOutputSync"
```

セットアップを確認する

自動ダウンロードの設定が成功したことを確認するには、次の手順を実行します。

1. 新しいテストジョブを送信します。
2. 1つのスケジューラ間隔が完了するまで待ちます。この場合は5分です。
3. 新しい出力が自動的にダウンロードされることを確認します。

出力がダウンロードされない場合は、プロセスログのトラブルシューティングセクションを確認してください。

自動ダウンロードのトラブルシューティング

自動ダウンロードで問題が発生した場合は、以下を確認してください。

ストレージプロファイルの問題

- ログファイル[Errno 13] Permission deniedの [Errno 2] No such file or directoryや などのエラーは、ストレージプロファイルの欠落や設定ミスに関連している可能性があります。
- ダウンロードマシンが送信マシンと異なる場合に[ストレージプロファイル](#)を設定する方法については、「ストレージプロファイル」を参照してください。
- 同じマシンのダウンロードの場合は、`--ignore-storage-profiles`フラグを試してください。

ディレクトリのアクセス許可

- スケジューラサービスユーザーに以下があることを確認します。
 - チェックポイントディレクトリへの読み取り/書き込みアクセス
 - 出力先ディレクトリへの書き込みアクセス
- Linux および の場合macOS、`ls -la` を使用してアクセス許可を確認します。
- についてはWindows、「プロパティ」フォルダのセキュリティ設定を確認します。

スケジューラログの確認

Linux

1. cron サービスが実行されているかどうかを確認します。

```
# For systemd systems
$ sudo systemctl status cron
# or
$ sudo systemctl status crond

# Check if your user has cron job correctly configured
$ crontab -l
```

2. cron 実行ログを表示します。

```
# Check system logs for cron activity (most common locations)
```

```
$ sudo tail -f /var/log/syslog | grep CRON
$ sudo tail -f /var/log/cron.log | grep deadline

# View recent cron logs
$ sudo journalctl -u cron -f
$ sudo journalctl -u crond -f # On some systems
```

3. 特定の cron ジョブログを確認します。

```
# View the log file specified in your cron job
$ tail -100f /path/to/logs/deadline_sync.log
```

4. システムログで cron ジョブ実行を検索します。

```
# Look for your specific cron job executions
$ sudo grep "deadline.*incremental-output-download" /var/log/syslog

# Check for cron job starts and completions
$ sudo grep "$(whoami).*CMD.*deadline" /var/log/syslog
```

5. チェックポイントファイルの更新を確認します。

```
# List checkpoint files with timestamps
$ ls -la /path/to/checkpoint/directory/

# Check when checkpoint was last modified
$ stat /path/to/checkpoint/directory/queue-*_download_checkpoint.json
```

6. ログファイルを確認します。

```
$ ls -la /path/to/log/deadline_sync.log
```

macOS

起動エージェント実行ログの表示:

1. 起動エージェントが実行中かどうかを確認します。

```
$ sudo launchctl list | grep deadlinesync
```

出力には以下が表示されます。PID Status Label (PID は、現在実行中で-ない場合があります。これは間隔ジョブでは正常です)

2. 詳細な起動エージェントのステータスを表示します。

```
$ sudo launchctl print system/com.user.deadlinesync
```

実行履歴、最後の終了コード、実行数、現在の状態が表示されます。

3. 起動エージェント実行ログを表示します。

```
# View recent logs (last hour)
log show --predicate 'subsystem contains "com.user.deadlinesync"' --last 1h

# View logs from a specific time period
log show --predicate 'subsystem contains "com.user.deadlinesync"' --start
'2024-08-27 09:00:00'
```

4. 即時テストのために起動エージェントを強制実行します。

```
$ sudo launchctl kickstart gui/${id -u}/com.user.deadlinesync
```

これにより、スケジュールに関係なくジョブがすぐにトリガーされ、テストに役立ちます。

5. チェックポイントファイルの更新を確認します。

```
# List checkpoint files with timestamps
$ ls -la /path/to/checkpoint/directory/
```

6. ログファイルを確認します。

```
$ ls -la /path/to/log/deadline_sync.log
```

Windows

1. Task Scheduler サービスが実行されているかどうかを確認します。

```
C:\> sc query "Schedule"
```

サービスが存在しない場合は、代替名を試してください。

```
C:\> sc query "TaskScheduler"
C:\> sc query "Task Scheduler"
```

2. スケジュールされたタスクを表示します。

```
C:> schtasks /query /tn "DeadlineOutputSync"
```

3. タスクのログファイルを確認します。

```
# View the log file created by your batch script
C:> notepad C:\path\to\logs\deadline_sync.log
```

4. チェックポイントファイルの更新を確認します。

```
# List checkpoint files with timestamps
```

```
C:> dir "C:\path\to\checkpoint\directory" /od
```

Deadline Cloud フォームの支出と使用状況を追跡する

AWS Deadline Cloud 予算マネージャーと使用状況エクスペローラーは、コスト変数に関する利用可能な情報に基づいて Deadline Cloud を使用するおおよそのコストを提供するコスト管理ツールです。コスト管理ツールは、Deadline Cloud およびその他の AWS サービスの実際の使用に対して支払うべき金額を保証するものではありません。

Deadline Cloud のコスト管理に役立つように、次の機能を使用できます。

- 予算マネージャー – Deadline Cloud 予算マネージャーを使用すると、プロジェクトコストの管理に役立つ予算を作成および編集できます。
- Usage Explorer – Deadline Cloud Usage Explorer を使用すると、使用されている AWS リソースの数とそれらのリソースの推定コストを確認できます。
- コストスケール係数 – コストスケール係数を使用すると、使用量エクスペローラーと予算マネージャーにコストを表示する方法を調整して、組織に適用される割引やプレミアムを反映することができます。
- AWS コスト配分タグ – コスト配分タグを使用すると、すべての AWS サービスの詳細なコストを追跡できます。詳細については、[「コスト配分タグを使用した AWS コストの整理と追跡」](#)を参照してください。

コストの前提

Deadline Cloud コスト管理ツールで使用される基本的な計算は次のとおりです。

```
Cost per job =  
  (CMF run time x CMF compute rate) +  
  (SMF run time x SMF compute rate) +  
  (License run time x license rate)
```

- 実行時間は、開始時刻から終了時刻までのジョブ内のすべてのタスクの合計です。
- コンピューティングレートは、サービスマネージドフリートの [AWS Deadline Cloud 料金](#)によって決まります。カスターマネージドフリートの場合、コンピューティングレートはワーカー 1 時間あたり 1 USD と推定されます。
- ライセンスレートは Deadline Cloud の基本ライセンス料金によって決定され、サービスマネージドフリートでのみ使用できます。追加の階層は含まれません。ライセンス料金の詳細については、[AWS 「Deadline Cloud の料金」](#)を参照してください。

Deadline Cloud コスト管理ツールからのコスト見積もりは、さまざまな理由で実際のコストとは異なる場合があります。一般的な理由は次のとおりです。

- 顧客所有のリソースとその料金。独自のリソースをオンプレミス AWS や他のクラウドプロバイダーから持ち込むか、外部から持ち込むかを選択できます。これらのリソースの実際のコストは計算されません。
- アイドルワーカーのコスト。ワーカーのステータスが IDLE の場合、アイドルワーカーのコストは含まれません。この状況は、最小インスタンス数が 0 より大きいフリート、またはワーカーがジョブ間で移行するときに発生する可能性があります。アイドルワーカーのコストは計算に含まれません。
- ワーカーの停止時刻と開始時刻。ワーカーがジョブを完了すると、IDLE から STOPPING に移行し、STOPPING から STOPPED に移行するためのコストは、Deadline Cloud のコスト見積もりに含まれません。
- プロモーションクレジット、割引、カスタム料金契約。コスト管理ツールは、プロモーションクレジット、プライベート料金契約、またはその他の割引を考慮しません。見積りに含まれない他の割引の対象となる場合があります。これらの要因を反映するように表示コストを調整するには、[を使用しますコストスケール係数](#)。
- アセットストレージ。アセットストレージは、コストと使用量の見積もりに含まれません。
- price. AWS offers でのほとんどのサービスの従量制料金の変更。pay-as-you-go 料金は時間の経過とともに変更される可能性があります。コスト管理ツールは、公開up-to-date最新の料金を使用しますが、変更後に遅延が発生する場合があります。
- 税金。コスト管理ツールには、サービスの購入に適用される税金は含まれません。
- 四捨五入。コスト管理ツールは、料金データの数学的四捨五入を実行します。
- 通貨。コスト見積もりは米ドルで行われます。グローバル為替レートは、時間の経過とともに変化します。見積りを現在の交換に基づいて別の通貨ベースに変換すると、為替レートの変更が見積りに影響します。
- 外部ライセンス。事前に購入したライセンス ([サービスマネージドフリートのソフトウェアライセンス](#)) を使用する場合、Deadline Cloud コスト管理ツールはこのコストを考慮できません。

コストスケール係数

コストスケール係数は、使用量エクスペローラーと予算マネージャーに表示される計算コストに乗数を適用するファームレベルの設定です。コストスケール係数を使用して、プライベート料金契約、プロモーションクレジット、内部コスト配分マークアップなど、コスト見積もりを組織の実際の料金に合わせます。

コストスケール係数値

コストスケール係数は 0～100 の値を受け入れます。

- 1 未満の値は割引を表します。たとえば、値が 0.75 の場合、表示されるコストに 25% の割引が適用されます。
- 1 より大きい値は、プレミアムまたはマークアップを表します。例えば、1.5 の値は、表示されたコストに 50% のマークアップを適用します。
- 値 1 (デフォルト) の場合、コストは変更されません。

コストスケール係数を設定する

ファームを作成するとき、または既存のファームの設定を編集することで、コストスケール係数を設定できます。

既存のファームのコストスケール係数を設定するには

1. [AWS Deadline Cloud \(Deadline Cloud\) コンソール](#)を開きます。ナビゲーションペインで、ファームやその他のリソースを選択します。
2. 変更するファームを選択します。
3. [アクション] をクリックして、[編集] を選択します。
4. コストスケール係数には、0～100 の値を入力します。
5. [Save changes] (変更の保存) をクリックします。

コストスケール係数がコストツールに与える影響

コストスケール係数を設定すると、その値は使用量エクスペローラーと予算マネージャーに次の方法で影響します。

- Usage Explorer – すべての新しいクエリには、コストスケール係数によって変更されたコストデータが表示されます。
- 新しい予算 – コストスケール係数を設定した後に作成された予算は、すべてのコスト計算に新しい値を使用します。
- 既存の予算 – 既存の予算は新しいコスト計算にコストスケール係数を使用しますが、累積コスト履歴は再計算されません。新しい要素を使用して累積コストを再計算するには、予算を削除して再作成します。

予算によるコストの管理

Deadline Cloud 予算マネージャーは、キュー、フリート、ファームなど、特定のリソースに対する支出を制御するのに役立ちます。予算の金額と制限を作成し、予算に対する追加支出を削減または停止するのに役立つ自動アクションを設定できます。

以下のセクションでは、Deadline Cloud 予算マネージャーを使用する手順について説明します。

トピック

- [前提条件](#)
- [Deadline Cloud 予算マネージャーを開く](#)
- [Deadline Cloud キューの予算を作成する](#)
- [Deadline Cloud キューの予算を表示する](#)
- [Deadline Cloud キューの予算を編集する](#)
- [Deadline Cloud キューの予算を無効にする](#)
- [EventBridge イベントで予算をモニタリングする](#)

前提条件

Deadline Cloud 予算マネージャーを使用するには、OWNERアクセスレベルが必要です。アクセスOWNER許可を付与するには、「」の手順に従います[Deadline Cloud でのユーザーの管理](#)。

Deadline Cloud 予算マネージャーを開く

Deadline Cloud 予算マネージャーを開くには、次の手順を使用します。

1. にサインイン AWS マネジメントコンソール し、Deadline Cloud [コンソール](#)を開きます。
2. ファームの表示を選択します。
3. 情報を取得するファームを見つけ、ジョブの管理を選択します。
4. Deadline Cloud モニターの左側のナビゲーションペインで、Budgets を選択します。

予算マネージャーの概要ページには、アクティブな予算と非アクティブな予算の両方のリストが表示されます。

- アクティブな予算は、選択したリソース (キュー) に対して追跡されます。

- 非アクティブな予算の有効期限が切れているか、ユーザーによってキャンセルされ、この予算の制限に対してコストを追跡しなくなりました。

予算を選択すると、予算の概要ページに予算に関する基本情報が表示されます。提供される情報には、予算名、ステータス、リソース、残りの割合、残りの金額、合計予算、開始日、終了日が含まれます。

Deadline Cloud キューの予算を作成する

予算を作成するには、次の手順を使用します。

1. まだサインインしていない場合は、にサインインし AWS マネジメントコンソール、Deadline Cloud [コンソール](#)を開き、ファームを選択し、ジョブの管理を選択します。
2. Budget manager ページで、Create budget を選択します。
3. 詳細セクションに、予算の予算名を入力します。
4. (オプション) 説明フィールドに、予算の簡単な説明を入力します。
5. リソースから、キュードロップダウンを使用して、予算を作成するキューを選択します。
6. Period では、次のステップを実行して、予算の開始日と終了日を設定します。
 - a. 開始日には、予算追跡の最初の日付を YYYY/MM/DD 形式で入力するか、カレンダーアイコンを選択して日付を選択します。

デフォルトの開始日は、予算が作成された日付です。
 - b. 終了日には、予算追跡の最終日を YYYY/MM/DD 形式で入力するか、カレンダーアイコンを選択して日付を選択します。

デフォルトの終了日は、開始日から 120 日です。
7. 予算額には、予算のドル額を入力します。
8. (オプション) 制限アラートを作成することをお勧めします。アクションの制限セクションでは、特定の金額が予算に残ったときに発生する自動アクションを実装できます。そのためには、以下のステップを完了します。
 - a. 新しいアクションの追加 を選択します。
 - b. 残りの金額には、アクションを開始するドル金額を入力します。
 - c. アクションドロップダウンで、必要なアクションを選択します。アクションには以下が含まれます。

- 現在の作業を終了した後で停止 – しきい値に達したときに現在実行中のすべての作業は、完了するまで引き続き実行されます (コストが発生します)。
 - 作業の即時停止 – しきい値に達すると、すべての作業が直ちにキャンセルされます。
- d. 追加の制限アラートを作成するには、新しいアクションを追加を選択し、前のステップを繰り返します。
9. [予算を作成] をクリックします。

Deadline Cloud キューの予算を表示する

予算を作成したら、予算マネージャーページで予算を表示できます。そこから、予算の合計金額と、特定の予算に割り当てられた全体的なコストを表示できます。

予算を表示するには、次の手順を使用します。

1. まだの場合は、 にサインインし AWS マネジメントコンソール、Deadline Cloud [コンソール](#)を開き、ファームを選択し、ジョブの管理を選択します。
2. 左側のナビゲーションペインから Budgets を選択します。Budget Manager ページが表示されます。
3. アクティブな予算を表示するには、アクティブな予算タブを選択し、表示する予算の名前を選択します。予算の詳細ページが表示されます。
4. 期限切れの予算の予算の詳細を表示するには、非アクティブな予算タブを選択します。次に、表示する予算の名前を選択します。予算の詳細ページが表示されます。

Deadline Cloud キューの予算を編集する

アクティブな予算は編集できます。アクティブな予算を編集するには、次の手順を使用します。

1. まだの場合は、 にサインインし AWS マネジメントコンソール、Deadline Cloud [コンソール](#)を開き、ファームを選択し、ジョブの管理を選択します。
2. Budget Manager ページのアクティブ予算タブで、編集する予算の横にあるボタンを選択します。
3. Actions ドロップダウンメニューから、Edit budget を選択します。
4. 必要な変更を行い、予算の更新を選択します。

Deadline Cloud キューの予算を無効にする

アクティブな予算は非アクティブ化できます。予算を非アクティブ化すると、そのステータスがアクティブから非アクティブに変更されます。予算が非アクティブ化されると、その予算の金額までリソースを追跡しなくなります。

予算を非アクティブ化するには、次の手順を使用します。

1. まだサインインしていない場合は、にサインインし AWS マネジメントコンソール、Deadline Cloud [コンソール](#)を開き、ファームを選択し、ジョブの管理を選択します。
2. Budget Manager ページの Active Budgets タブで、非アクティブ化する予算の横にあるボタンを選択します。
3. Actions ドロップダウンメニューから、Deactivate budget を選択します。しばらくすると、選択した予算がアクティブから非アクティブに変更され、アクティブ予算タブから非アクティブ予算タブに移動します。

EventBridge イベントで予算をモニタリングする

Deadline Cloud は、Amazon EventBridge を使用して予算関連のイベントをデフォルトの EventBridge イベントバスに送信します。イベントを受信し、それに基づいて通知を送信するカスタム関数を作成して、予算が事前定義されたレベルに達したときに、E メール、Slack、またはその他のチャンネルを介してユーザーに自動的に通知できます。たとえば、予算が特定のしきい値に達したときに SMS メッセージを送信できます。これらの通知は、支出を把握し、予算を使い果たす前に情報に基づいた意思決定を行うのに役立ちます。

Deadline Cloud は、各レンダーファームの使用状況とコストデータを定期的に集計します。次に、予算しきい値のいずれかが超過したかどうかを確認します。しきい値を超えると、Deadline Cloud はイベントをトリガーして警告し、適切なアクションを実行できるようにします。予算がこれらのしきい値の 1 つを超えると、イベントがトリガーされ、使用される予算の割合で指定されます。

- 10、20、30、40、50、60、70、75、80、85、90、95、96、97、98、99、100

予算使用量のしきい値は、予算が 100% の使用に近づくとつれて近づきます。この頻度は、予算が制限に達するにつれて使用状況を注意深くモニタリングするのに役立ちます。独自の予算しきい値を設定することもできます。Deadline Cloud は、使用量がカスタムしきい値を超えるとイベントを送信します。予算が 100% に達すると、Deadline Cloud はイベントの送信を停止します。予算を調整すると、Deadline Cloud は新しい予算額に基づいてしきい値のイベントを送信します。

EventBridge コンソール (<https://console.aws.amazon.com/events/>) を使用して、Deadline Cloud イベントをイベントの適切なターゲットに送信するルールを作成できます。たとえば、イベントを Amazon Simple Queue Service キューに送信し、そこから AWS End User Messaging SMS や Amazon Relational Database Service データベースなどの複数のターゲットに送信してログを記録できます。

EventBridge ルールの例については、以下のトピックを参照してください。

- [Amazon EventBridge を使用してイベントが発生したときに E メールを送信します。](#)
- [チャットアプリケーションで Amazon Q Developer に通知を送信する Amazon EventBridge ルールを作成します。](#)
- [Amazon EventBridge の開始方法。](#)

予算イベントの詳細については、Deadline Cloud デベロッパーガイドの [Budget Threshold Reached イベント](#) を参照してください。

Deadline Cloud 使用状況エクスペローラーを使用して使用状況とコストを追跡する

Deadline Cloud 使用状況エクスペローラーを使用すると、各ファームで発生しているアクティビティに関するリアルタイムのメトリクスを確認できます。ファームのコストは、キュー、フリート、ジョブ、ライセンス製品、インスタンスタイプなど、さまざまな変数別に確認できます。さまざまな時間枠を選択して、特定の期間における使用状況を確認し、その期間における使用状況の傾向を確認します。選択したデータポイントの詳細な内訳を表示して、メトリクスを詳しく確認することもできます。使用状況は、時間 (分と時間) またはコスト (\$USD) で表示できます。

以下のセクションでは、Deadline Cloud 使用状況エクスペローラーにアクセスして使用する手順を示します。

トピック

- [前提条件](#)
- [使用量エクスペローラーを開く](#)
- [使用量エクスペローラーを使用する](#)

前提条件

Deadline Cloud 使用状況エクスペローラーを使用するには、MANAGERまたはOWNERファームのアクセス許可が必要です。詳細については、「[アクセスレベルについて](#)」を参照してください。

Note

タイムゾーンがインド標準時 (UTC+5:30) などの 1 時間と一致しない場合、使用状況エクスペローラーは使用状況メトリクスを表示しません。メトリクスを表示するには、タイムゾーンを 1 時間と一致するゾーンに設定します。

使用量エクスペローラーを開く

Deadline Cloud 使用状況エクスペローラーを開くには、次の手順を使用します。

1. にサインイン AWS マネジメントコンソールし、Deadline Cloud [コンソール](#)を開きます。
2. 使用可能なすべてのファームを表示するには、ファームの表示を選択します。
3. 情報を取得するファームを見つけ、ジョブの管理を選択します。Deadline Cloud モニターが新しいタブで開きます。
4. Deadline Cloud モニターの左側のメニューから、Usage Explorer を選択します。

使用量エクスペローラーを使用する

使用状況エクスペローラーページから、データを表示できる特定のパラメータを選択できます。デフォルトでは、過去 7 日間の時間 (時と分) の合計使用量が表示されます。これらのパラメータを変更でき、表示される情報はパラメータ設定に従って動的に変わります。

結果は、キュー、フリート、ジョブ、ユーザー、コンピューティング使用量、インスタンスタイプ、ライセンス製品に基づいてグループ化できます。ライセンス製品を選択した場合、コストは特定のライセンスに対して計算されます。他のすべてのグループについては、各タスクの実行にかかる時間を合計して時間が計算されます。

結果はキューまたはフリートでフィルタリングできますが、両方で同時にフィルタリングすることはできません。

使用量エクスペローラーは、設定したフィルター条件に基づいて 100 の結果のみを返します。結果は、作成されたタイムスタンプの降順に表示されます。結果が 100 を超える場合は、エラーメッセージが表示されます。クエリを絞り込んで、結果の数を減らすことができます。

- より小さい時間範囲を選択する
- 選択するキューまたはフリートの数を減らす
- ジョブではなくキューまたはフリートによるグループ化など、別のグループ化を選択する

トピック

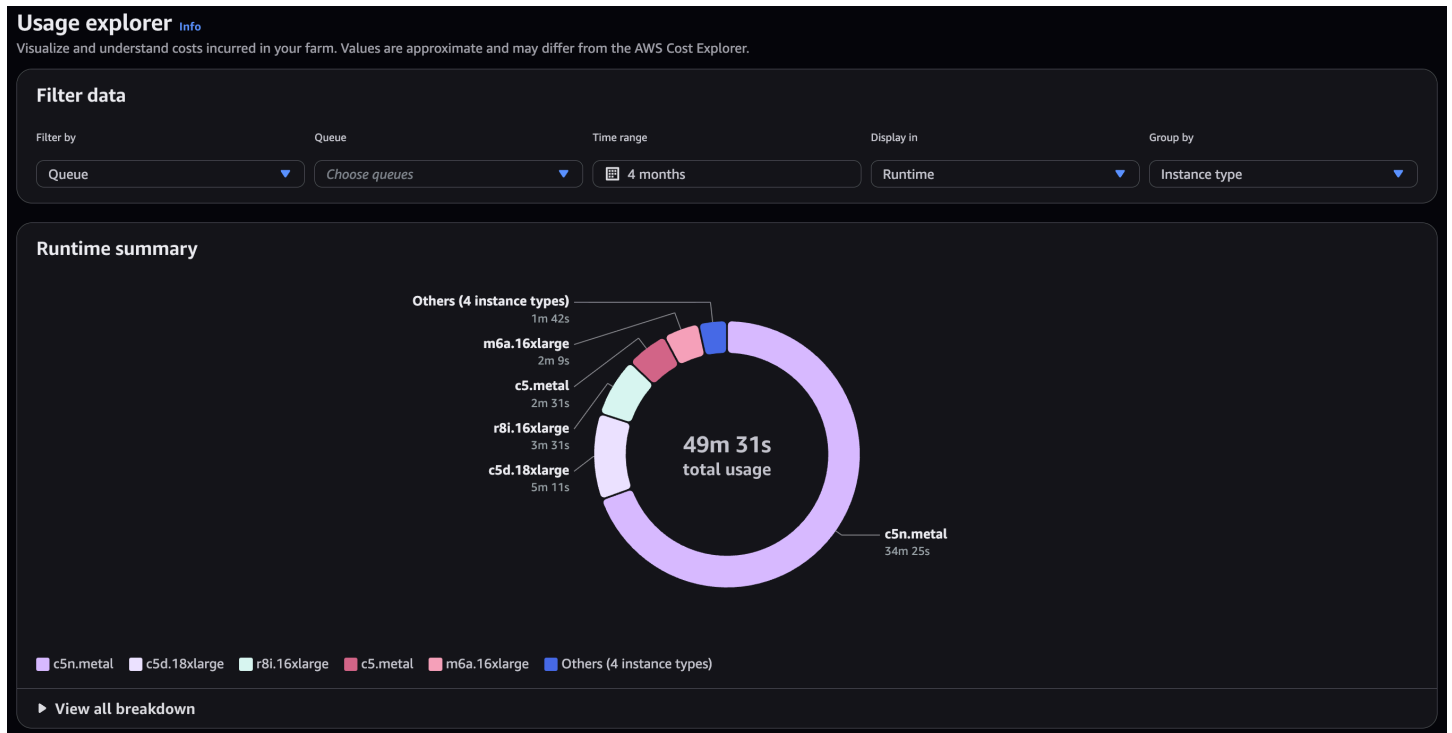
- [ビジュアルグラフを使用してデータを確認する](#)
- [メトリクスの内訳を表示する](#)
- [キューとフリートのおおよそのランタイムを表示する](#)

ビジュアルグラフを使用してデータを確認する

データを視覚的な形式で確認して、より多くの分析や注意が必要な傾向や潜在的な領域を特定できます。Usage Explorer には、全体的な使用量とコストを表示する円グラフがあり、合計を小さな小計にグループ化するオプションがあります。

Note

グラフには、上位 5 つの結果のみが表示され、他の結果が「その他」セクションに結合されます。すべての結果は、グラフの下の内訳セクションで表示できます。



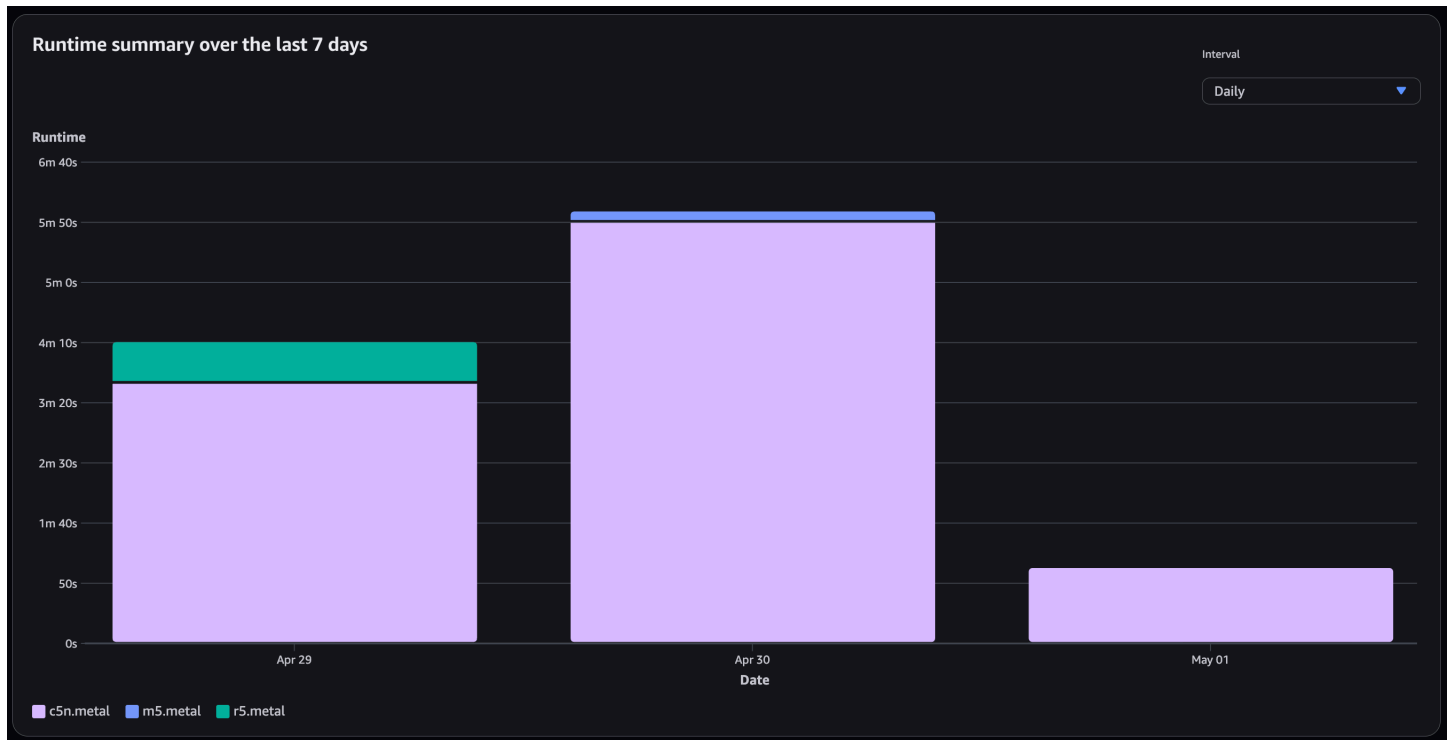
メトリクスの内訳を表示する

円グラフの下には、特定のメトリクスのより詳細な内訳が表示されます。これはパラメータの変化に応じて変化します。デフォルトでは、Usage Explorer に 5 つの結果が表示されます。内訳セクションのページ分割矢印を使用して結果をスクロールできます。

デフォルトでは、内訳は最小限に抑えられます。結果を展開して表示するには、すべての内訳を表示矢印を選択します。内訳をダウンロードするには、データのダウンロードを選択します。

キューとフリートのおおよそのランタイムを表示する

指定したさまざまな間隔に基づいて、キューまたはフリートのおおよそのランタイムを表示することもできます。間隔オプションは、時間単位、日単位、週単位、月単位です。間隔を選択すると、グラフにはキューまたはフリートのおおよそのランタイムが表示されます。



コスト管理

AWS Deadline Cloud は、ジョブのコストを制御および視覚化するのに役立つ予算と使用状況エクスペローラーを提供します。ただし、Deadline Cloud は Amazon S3 などの他の AWS サービスを使用します。これらのサービスのコストは Deadline Cloud 予算や Usage Explorer には反映されず、使用量に基づいて個別に課金されます。Deadline Cloud の設定方法によっては、以下の AWS サービスだけでなく、その他のサービスも使用できます。

サービス	料金表ページ
Amazon CloudWatch Logs	Amazon CloudWatch Logs の料金
Amazon Elastic Compute Cloud	Amazon Elastic Compute Cloud の料金
AWS Key Management Service	AWS Key Management Service 料金表
AWS PrivateLink	AWS PrivateLink 料金表
Amazon Simple Storage Service	Amazon Simple Storage Service の料金表
Amazon Virtual Private Cloud	Amazon Virtual Private Cloud の料金

コスト管理のベストプラクティス

次のベストプラクティスを使用すると、Deadline Cloud を使用する際のコストと、コストと効率の間のトレードオフを理解して制御できます。

Note

Deadline Cloud を使用するための最終的なコストは、多数の AWS サービス間のやり取り、処理する作業量、ジョブを実行する AWS リージョン によって異なります。以下のベストプラクティスはガイドラインであり、大幅なコスト削減にはつながらない場合があります。

CloudWatch Logs のベストプラクティス

Deadline Cloud は、ワーカーログとタスクログを CloudWatch Logs に送信します。これらのログの収集、保存、分析には料金が発生します。タスクのモニタリングに必要な最小限のデータのみをログに記録することで、コストを削減できます。

キューまたはフリートを作成すると、Deadline Cloud は次の名前の CloudWatch Logs ロググループを作成します。

- /aws/deadline/<FARM_ID>/<FLEET_ID>
- /aws/deadline/<FARM_ID>/<QUEUE_ID>

デフォルトでは、これらのログには有効期限はありません。ロググループの保持ポリシーを調整して古いログを削除し、ストレージコストを削減できます。ログを Simple Storage Service (Amazon S3) にエクスポートすることもできます。Amazon S3 のストレージコストは、CloudWatch のストレージコストよりも低くなります。詳細については、「[Amazon S3 へのログデータのエクスポート](#)」を参照してください。

Amazon EC2 のベストプラクティス

Amazon EC2 インスタンスは、サービスマネージドフリートとカスターマネージドフリートの両方に使用できます。次の 3 つの考慮事項があります。

- サービスマネージドフリートの場合、フリートの最小ワーカー数を設定することで、1 つ以上のインスタンスを常に使用可能にすることができます。最小ワーカー数を 0 に設定すると、フリートは常にこの数のワーカーを実行します。この設定により、Deadline Cloud がジョブの処理を開始するのにかかる時間を短縮できますが、インスタンスのアイドル時間に対して課金されます。

- サービスマネージドフリートの場合は、フリートの最大サイズを設定します。この設定は、フリートが自動スケーリングできるインスタンスの数を制限します。処理を待っているジョブが他にもあっても、フリートがこのサイズを超えることはありません。
- サービスマネージドフリートとカスターマネージドフリートの両方で、フリートで Amazon EC2 インスタンスタイプを指定できます。小規模なインスタンスを使用すると、1分あたりのコストは削減されますが、ジョブの完了に時間がかかる場合があります。逆に、インスタンスが大きいほど1分あたりのコストは高くなりますが、ジョブを完了する時間を短縮できます。ジョブがインスタンスに配置する需要を理解することで、コストを削減できます。
- 可能であれば、フリートの Amazon EC2 スポットインスタンスを選択します。スポットインスタンスは割引価格で利用できますが、オンデマンドリクエストによって中断される可能性があります。オンデマンドインスタンスは2秒ごとに課金され、中断されません。

のベストプラクティス AWS KMS

デフォルトでは、Deadline Cloud は AWS 所有キーを使用してデータを暗号化します。このキーには課金されません。

カスターマネージドキーを使用してデータを暗号化することもできます。独自のキーを使用する場合、キーの使用法に基づいて課金されます。既存のキーを使用する場合、これは追加使用の増分コストになります。

のベストプラクティス AWS PrivateLink

を使用して AWS PrivateLink、インターフェイスエンドポイントを使用して VPC と Deadline Cloud 間の接続を作成できます。接続を作成するときに、すべての Deadline Cloud API アクションを呼び出すことができます。作成したエンドポイントごとに1時間あたりに課金されます。PrivateLink を使用する場合は、少なくとも3つのエンドポイントを作成する必要があります。設定によっては、最大5つのエンドポイントが必要になる場合があります。

Amazon S3 のベストプラクティス

Deadline Cloud は Amazon S3 を使用して、処理、ジョブの添付ファイル、出力、ログ用のアセットを保存します。Amazon S3 に関連するコストを削減するには、保存するデータの量を減らします。いくつかの提案:

- 現在使用中のアセット、または間もなく使用されるアセットのみを保存します。
- [S3 ライフサイクル設定](#)を使用して、S3 バケットから未使用のファイルを自動的に削除します。

Amazon VPC のベストプラクティス

カスタマーマネージドフリートに使用状況ベースのライセンスを使用する場合は、アカウントで作成された Amazon VPC エンドポイントである Deadline Cloud ライセンスエンドポイントを作成します。このエンドポイントは 1 時間あたりの料金で課金されます。コストを削減するには、使用量ベースのライセンスを使用していない場合はエンドポイントを削除します。

のセキュリティ Deadline Cloud

でのクラウドセキュリティが最優先事項 AWS です。お客様は AWS、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、AWS とお客様の間の責任共有です。[責任共有モデル](#)ではこれをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – AWS は、AWS のサービス で実行されるインフラストラクチャを保護する責任を担います AWS クラウド。は、お客様が安全に使用できるサービス AWS も提供します。サードパーティーの監査者は、[AWS コンプライアンスプログラム](#)コンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証。が適用されるコンプライアンスプログラムの詳細については AWS Deadline Cloud、「[コンプライアンスAWS のサービス プログラムによる対象範囲内](#)」「コンプライアンス」を参照してください。AWS Deadline Cloud は SOC 1、2、および 3 のコンプライアンスの対象範囲内です。詳細については、「[the section called “コンプライアンス検証”](#)」を参照してください。
- クラウド内のセキュリティ – お客様の責任は、使用する によって決まり AWS のサービス ます。また、ユーザーは、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、 を使用する際の責任共有モデルの適用方法を理解するのに役立ちます Deadline Cloud。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成する Deadline Cloud ように を設定する方法を示します。また、Deadline Cloud リソースのモニタリングと保護 AWS のサービス に役立つ他の の使用方法についても説明します。

トピック

- [でのデータ保護 Deadline Cloud](#)
- [Deadline Cloud での Identity and Access Management](#)
- [のコンプライアンス検証 Deadline Cloud](#)
- [の耐障害性 Deadline Cloud](#)
- [Deadline Cloud のインフラストラクチャセキュリティ](#)
- [Deadline Cloud の設定と脆弱性の分析](#)
- [サービス間での不分別な代理処理の防止](#)

- [インターフェイスエンドポイント \(AWS PrivateLink\) AWS Deadline Cloud を使用した へのアクセス](#)
- [制限されたネットワーク環境](#)
- [Deadline Cloud のセキュリティのベストプラクティス](#)

でのデータ保護 Deadline Cloud

AWS [責任共有モデル](#) は、AWS Deadline Cloudでのデータ保護に適用されます。このモデルで説明されているように、AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「AWS のサービス」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「[Data Privacy FAQChina](#)」を参照してください。欧州におけるデータ保護に関する情報については、[General Data Protection Regulation \(GDPR\) Center](#) を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM アイデンティティセンターまたは AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「AWS CloudTrail ユーザーガイド」の[CloudTrail 証跡の使用](#)」を参照してください。
- AWS 暗号化ソリューションと、内のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済みの暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-3](#)」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して AWS CLI Deadline Cloud または他の AWS のサービス を操作する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

Deadline Cloud ジョブテンプレートの名前フィールドに入力されたデータは、請求ログや診断ログに含まれている可能性があるため、機密情報や機密情報を含めないでください。

トピック

- [保管中の暗号化](#)
- [転送中の暗号化](#)
- [キー管理](#)
- [ネットワーク間トラフィックのプライバシー](#)
- [オプトアウト](#)

保管中の暗号化

AWS Deadline Cloud は、[AWS Key Management Service \(AWS KMS\)](#) に保存されている暗号化キーを使用して保管時に暗号化することで、機密データを保護します。保管時の暗号化は、AWS リージョン Deadline Cloud が利用可能なすべてので使用できます。

データの暗号化とは、ディスクに保存された機密データが、有効なキーがないユーザーやアプリケーションによって読み取れないことを意味します。有効なマネージドキーを持つ当事者のみがデータを復号できます。

Deadline Cloud サービスマネージドフリートワーカーインスタンスが終了すると、は Amazon Elastic Block Store ボリュームを削除します。

が保管中のデータの暗号化 AWS KMS にどのように Deadline Cloud 使用されるかについては、「」を参照してください[キー管理](#)。

転送中の暗号化

転送中のデータの場合、AWS Deadline Cloud は Transport Layer Security (TLS) 1.2 または 1.3 を使用して、サービスとワーカー間で送信されるデータを暗号化します。TLS 1.2 は必須ですが、TLS

1.3 を推奨します。さらに、Virtual Private Cloud (VPC) を使用する場合は、AWS PrivateLink を使用して VPC と 間のプライベート接続を確立できます Deadline Cloud。

キー管理

新しいファームを作成するときは、次のいずれかのキーを選択してファームデータを暗号化できます。

- AWS 所有 KMS キー – ファームの作成時にキーを指定しない場合のデフォルトの暗号化タイプ。KMS キーは によって所有されています AWS Deadline Cloud。AWS 所有キーを表示、管理、使用することはできません。ただし、データを暗号化するキーを保護するためにアクションを実行する必要はありません。詳細については、「[デAWS Key Management Service ベロツパーガイド](#)」の [AWS 「所有キー」](#) を参照してください。
- カスタマーマネージド KMS キー – ファームの作成時にカスタマーマネージドキーを指定します。ファーム内のすべてのコンテンツは KMS キーで暗号化されます。キーはアカウントに保存され、ユーザーが作成、所有、管理し、AWS KMS 料金が適用されます。ユーザーは、KMS キーに関する完全なコントロール権を持ちます。次のようなタスクを実行できます。
 - キーポリシーの確立と維持
 - IAM ポリシーとグラントの策定と維持
 - キーポリシーの有効化と無効化
 - タグを追加する
 - キーエイリアスの作成

Deadline Cloud ファームで使用される顧客所有のキーを手動でローテーションすることはできません。キーの自動ローテーションがサポートされています。

詳細については、「[AWS Key Management Service デベロツパーガイド](#)」の [「カスタマー所有キー」](#) を参照してください。

カスタマーマネージドキーを作成するには、「[AWS Key Management Service デベロツパーガイド](#)」の [「対称カスタマーマネージドキーの作成」](#) の手順に従います。

Deadline Cloud が AWS KMS 許可を使用する方法

Deadline Cloud には、カスタマーマネージドキーを使用するための [許可](#) が必要です。カスタマーマネージドキーで暗号化されたファームを作成すると、 は、指定した KMS キーへのアクセスを取得す

る [CreateGrant](#) リクエスト AWS KMS を に送信して、ユーザーに代わって許可 Deadline Cloud を作成します。

Deadline Cloud は複数の許可を使用します。各権限は、データを暗号化または復号 Deadline Cloud する必要がある の異なる部分によって使用されます。Deadline Cloud また、 は権限を使用して、Amazon Simple Storage Service、Amazon Elastic Block Store、OpenSearch など、ユーザーに代わってデータを保存するために使用される他の AWS サービスへのアクセスを許可します。

がサービスマネージドフリート内のマシンを管理 Deadline Cloud できるようにする権限には、Deadline Cloud サービスプリンシパル `GranteePrincipal` の代わりに のアカウント番号とロールが含まれます。これは一般的ではありませんが、ファームに指定されたカスターマネージド KMS キーを使用して、サービスマネージドフリートのワーカーの Amazon EBS ボリュームを暗号化するために必要です。

カスターマネージドキーポリシー

キーポリシーは、カスターマネージドキーへのアクセスを制御します。各キーには、キーを使用できるユーザーとその使用方法を決定するステートメントを含むキーポリシーが 1 つだけ必要です。カスターマネージドキーを作成するときに、キーポリシーを指定できます。詳細については、AWS Key Management Service デベロッパーガイドの「[Managing access to customer managed keys](#)」を参照してください。

CreateFarm の最小 IAM ポリシー

カスターマネージドキーを使用して コンソールまたは [CreateFarm](#) API オペレーションを使用してファームを作成するには、次の AWS KMS API オペレーションを許可する必要があります。

- [kms:CreateGrant](#) - カスターマネージドキーに許可を追加します。指定された AWS KMS キーへのコンソールアクセスを許可します。詳細については、「[デAWS Key Management Service ベロツパーガイド](#)」の「[許可の使用](#)」を参照してください。
- [kms:Decrypt](#) - Deadline Cloud がファーム内のデータを復号できるようにします。
- [kms:DescribeKey](#) - がキー Deadline Cloud を検証できるように、カスターマネージドキーの詳細を提供します。
- [kms:GenerateDataKey](#) - が一意のデータキーを使用してデータを暗号化 Deadline Cloud できるようにします。

次のポリシーステートメントは、CreateFarmオペレーションに必要なアクセス許可を付与します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineCreateGrants",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/1234567890abcdef0",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "deadline.us-west-2.amazonaws.com"
        }
      }
    }
  ]
}
```

読み取り専用オペレーションの最小 IAM ポリシー

ファーム、キュー、フリートに関する情報の取得など、カスタマーマネージドキーを読み取り専用 Deadline Cloud オペレーションに使用するには。次の AWS KMS API オペレーションを許可する必要があります。

- [kms:Decrypt](#) – Deadline Cloud がファーム内のデータを復号できるようにします。
- [kms:DescribeKey](#) – がキー Deadline Cloud を検証できるように、カスタマーマネージドキーの詳細を提供します。

次のポリシーステートメントは、読み取り専用オペレーションに必要なアクセス許可を付与します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadOnly",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "deadline.us-west-2.amazonaws.com"
        }
      }
    }
  ]
}
```

読み取り/書き込みオペレーションの最小 IAM ポリシー

ファーム、キュー、フリートの作成や更新などの読み取り/書き込み Deadline Cloud オペレーションにカスタマーマネージドキーを使用するには。次の AWS KMS API オペレーションを許可する必要があります。

- [kms:Decrypt](#) – Deadline Cloud がファーム内のデータを復号できるようにします。
- [kms:DescribeKey](#) – がキー Deadline Cloud を検証できるように、カスタマーマネージドキーの詳細を提供します。
- [kms:GenerateDataKey](#) – が一意のデータキーを使用してデータを暗号化 Deadline Cloud できるようにします。

次のポリシーステートメントは、CreateFarmオペレーションに必要なアクセス許可を付与します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadWrite",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:GenerateDataKey"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "deadline.us-west-2.amazonaws.com"
        }
      }
    }
  ]
}
```

暗号化キーのモニタリング

Deadline Cloud フォームで AWS KMS カスタマーマネージドキーを使用する場合、[AWS CloudTrail](#)または [Amazon CloudWatch Logs](#) を使用して、 が Deadline Cloud に送信するリクエストを追跡できます AWS KMS。

許可の CloudTrail イベント

次の CloudTrail イベント例は、通常、CreateFarm、または CreateFleetオペレーションを呼び出すときにCreateMonitor、許可が作成されたときに発生します。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
```

```
"accountId": "111122223333",
"accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
"sessionContext": {
  "sessionIssuer": {
    "type": "Role",
    "principalId": "AROAIQDTESTANDEXAMPLE",
    "arn": "arn:aws::iam::111122223333:role/Admin",
    "accountId": "111122223333",
    "userName": "Admin"
  },
  "webIdFederationData": {},
  "attributes": {
    "creationDate": "2024-04-23T02:05:26Z",
    "mfaAuthenticated": "false"
  }
},
"invokedBy": "deadline.amazonaws.com",
},
"eventTime": "2024-04-23T02:05:35Z",
"eventSource": "kms.amazonaws.com",
"eventName": "CreateGrant",
"awsRegion": "us-west-2",
"sourceIPAddress": "deadline.amazonaws.com",
"userAgent": "deadline.amazonaws.com",
"requestParameters": {
  "operations": [
    "CreateGrant",
    "Decrypt",
    "DescribeKey",
    "Encrypt",
    "GenerateDataKey"
  ],
  "constraints": {
    "encryptionContextSubset": {
      "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
      "aws:deadline:accountId": "111122223333"
    }
  },
  "granteePrincipal": "deadline.amazonaws.com",
  "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "retiringPrincipal": "deadline.amazonaws.com"
},
"responseElements": {
```

```
    "grantId": "6bbe819394822a400fe5e3a75d0e9ef16c1733143fff0c1fc00dc7ac282a18a0",
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
  },
  "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
  "readOnly": false,
  "resources": [
    {
      "accountId": "AWS Internal",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE44444"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}
```

復号用の CloudTrail イベント

次の CloudTrail イベント例は、カスタマーマネージド KMS キーを使用して値を復号するときに発生します。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
      },
      "webIdFederationData": {},

```

```
    "attributes": {
      "creationDate": "2024-04-23T18:46:51Z",
      "mfaAuthenticated": "false"
    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T18:51:44Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "Decrypt",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "deadline.amazonaws.com",
  "userAgent": "deadline.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {
      "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
      "aws:deadline:accountId": "111122223333",
      "aws-crypto-public-key": "AotL+SAMPLEVALUEiOMEXAMPLEEaaqNOTREALaGTESTONLY  
+p/5H+EuKd4Q=="
    },
    "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111"
  },
  "responseElements": null,
  "requestID": "aaaaaaaa-bbbb-cccc-dddd-eeeeefffffff",
  "eventID": "ffffffff-eeee-dddd-cccc-bbbbbbaaaaaa",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}
```

暗号化用の CloudTrail イベント

次の CloudTrail イベント例は、カスタマーマネージド KMS キーを使用して値を暗号化するときに発生します。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-04-23T18:46:51Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T18:52:40Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "deadline.amazonaws.com",
  "userAgent": "deadline.amazonaws.com",
  "requestParameters": {
    "numberOfBytes": 32,
    "encryptionContext": {
      "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
      "aws:deadline:accountId": "111122223333",
      "aws-crypto-public-key": "AotL+SAMPLEVALUEi0MEXAMPLEEaaqNOTREALaGTESTONLY
+p/5H+EuKd4Q=="
    }
  },
}
```

```
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/abcdef12-3456-7890-0987-654321fedcba"
  },
  "responseElements": null,
  "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}
```

カスタマーマネージド KMS キーの削除

AWS Key Management Service (AWS KMS) でカスタマーマネージド KMS キーを削除すると、破壊的であり、潜在的に危険です。これにより、キーマテリアルとキーに関連付けられているすべてのメタデータが削除され、元に戻すことはできません。カスタマーマネージド KMS キーを削除すると、そのキーで暗号化されたデータを復号できなくなります。キーを削除すると、データは回復できなくなります。

そのため、AWS KMS は KMS キーを削除する前に最大 30 日間の待機期間をお客様に付与します。デフォルトの待機時間は、30 日です。

待機期間について

カスタマーマネージド KMS キーを削除することは破壊的で潜在的に危険であるため、7~30 日間の待機期間を設定する必要があります。デフォルトの待機時間は、30 日です。

ただし、実際の待機期間は、スケジュールした期間よりも最大 24 時間長くなる場合があります。キーが削除される実際の日時を取得するには、[DescribeKey](#) オペレーションを使用します。また、[General configuration] (一般的な設定) セクションのキーの詳細ページにある [AWS KMS コンソール](#) では、削除のためにスケジュールされた日付を確認することが可能です。タイムゾーンに注意してください。

削除の待機期間中は、カスタマーマネージドキーのステータスおよびキーの状態が削除保留中になります。

- 削除保留中のカスタマーマネージド KMS キーは、[暗号化オペレーション](#)に使用することはできません。
- AWS KMS は、削除保留中のカスタマーマネージド KMS [キーのバックアップキーをローテーション](#)しません。

カスタマーマネージド KMS キーの削除の詳細については、AWS Key Management Service デベロッパーガイドの「[カスタマーマスターキーの削除](#)」を参照してください。

ネットワーク間トラフィックのプライバシー

AWS Deadline Cloud は Amazon Virtual Private Cloud (Amazon VPC) をサポートして接続を保護します。Amazon VPC は、Virtual Private Cloud (VPC) のセキュリティを強化、モニタリングするために使用できる機能を提供します。

VPC 内で実行される Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを使用して、カスタマーマネージドフリート (CMF) を設定できます。使用する Amazon VPC エンドポイントをデプロイすることで AWS PrivateLink、CMF のワーカーと Deadline Cloud エンドポイント間のトラフィックは VPC 内に留まります。さらに、インスタンスへのインターネットアクセスを制限するように VPC を設定できます。

サービスマネージドフリートでは、ワーカーはインターネットからアクセスできませんが、インターネットアクセスがあり、インターネット経由で Deadline Cloud サービスに接続できます。各サービスマネージドフリートは独自の独立したネットワークで実行され、ワーカーインスタンスは個々の顧客専用のままです。

オプトアウト

AWS Deadline Cloud は、開発と改善に役立つ特定の運用情報を収集します。Deadline Cloud。収集されたデータには、AWS アカウント ID やユーザー ID などの情報が含まれているため、に問題がある場合は正しく識別できます。Deadline Cloud。また、リソース IDs (該当する場合は FarmID または QueueID)、製品名 (JobAttachments、WorkerAgent など)、製品バージョンなどの Deadline Cloud 特定の情報を収集します。

アプリケーション設定を使用して、このデータ収集をオプトアウトできます。クライアントワークステーションとフリートワーカー Deadline Cloudの両方とやり取りする各コンピュータは、個別にオプトアウトする必要があります。

Deadline Cloud モニター - デスクトップ

Deadline Cloud monitor - デスクトップは、クラッシュが発生したときやアプリケーションが開かれたときなどの運用情報を収集し、アプリケーションに問題が発生したときの把握に役立ちます。この運用情報の収集をオプトアウトするには、設定ページに移動し、データ収集をオンにして Deadline Cloud Monitor のパフォーマンスを測定します。

オプトアウトすると、デスクトップモニターは運用データを送信しなくなります。以前に収集されたデータは保持され、引き続きサービスの改善に使用される可能性があります。詳細については、[データプライバシーのよくある質問](#)を参照してください。

AWS Deadline Cloud CLI とツール

AWS Deadline Cloud CLI、送信者、ワーカーエージェントはすべて、クラッシュが発生したときやジョブが送信されたときなどの運用情報を収集し、これらのアプリケーションで問題が発生したときの把握に役立ちます。この運用情報の収集をオプトアウトするには、次のいずれかの方法を使用します。

- ターミナルで、と入力します **deadline config set telemetry.opt_out true**。

これにより、現在のユーザーとして実行されているときに CLI、送信者、ワーカーエージェントがオプトアウトされます。

- Deadline Cloud ワーカーエージェントをインストールするときは、**--telemetry-opt-out** コマンドライン引数を追加します。例えば、 **./install.sh --farm-id \$FARM_ID --fleet-id \$FLEET_ID --telemetry-opt-out**。
- ワーカーエージェント、CLI、または送信者を実行する前に、環境変数を設定します。
DEADLINE_CLOUD_TELEMETRY_OPT_OUT=true

オプトアウトすると、Deadline Cloud ツールは運用データを送信しなくなります。以前に収集されたデータは保持され、引き続きサービスの改善に使用される可能性があります。詳細については、[データプライバシーのよくある質問](#)を参照してください。

Deadline Cloud での Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つです。IAM 管理者は、誰を認証 (サインイン) し、誰に Deadline

Cloud リソースの使用を許可する (アクセス許可を付与する) を制御します。IAM は、追加料金なしで使用できる AWS のサービスです。

トピック

- [オーディエンス](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [Deadline Cloud と IAM の連携方法](#)
- [Deadline Cloud のアイデンティティベースのポリシーの例](#)
- [AWS Deadline Cloud の マネージドポリシー](#)
- [サービスロール](#)
- [AWS Deadline Cloud のアイデンティティとアクセスのトラブルシューティング](#)

オーディエンス

AWS Identity and Access Management (IAM) の使用方法は、ロールによって異なります。

- サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします (「[AWS Deadline Cloud のアイデンティティとアクセスのトラブルシューティング](#)」を参照)。
- サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します (「[Deadline Cloud と IAM の連携方法](#)」を参照)
- IAM 管理者 - アクセスを管理するためのポリシーを作成します (「[Deadline Cloud のアイデンティティベースのポリシーの例](#)」を参照)

アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用してサインインする方法です。IAM ユーザー AWS アカウントのルートユーザー、または IAM ロールを引き受けることで認証される必要があります。

AWS IAM アイデンティティセンター (IAM Identity Center)、シングルサインオン認証、Google/Facebook 認証情報などの ID ソースからの認証情報を使用して、フェデレーテッド ID としてサインインできます。サインインの詳細については、「AWS サインイン ユーザーガイド」の「[AWS アカウントにサインインする方法](#)」を参照してください。

プログラムによるアクセスの場合、は SDK と CLI AWS を提供してリクエストを暗号化して署名します。詳細については、「IAM ユーザーガイド」の「[API リクエストに対するAWS 署名バージョン 4](#)」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、まず、すべての AWS のサービス および リソースへの完全なアクセス権を持つ AWS アカウント root ユーザーと呼ばれる 1 つのサインインアイデンティティから始めます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザー認証情報を必要とするタスクについては、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、人間のユーザーが一時的な認証情報 AWS のサービス を使用して にアクセスするには、ID プロバイダーとのフェデレーションを使用する必要があります。

フェデレーテッド ID は、エンタープライズディレクトリ、ウェブ ID プロバイダー、または ID Directory Service ソースの認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーテッドアイデンティティは、一時的な認証情報を提供するロールを引き受けます。

アクセスを一元管理する場合は、AWS IAM アイデンティティセンターをお勧めします。詳細については、「AWS IAM アイデンティティセンター ユーザーガイド」の「[IAM アイデンティティセンターとは](#)」を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、特定の個人やアプリケーションに対する特定のアクセス許可を持つアイデンティティです。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「[ID プロバイダーとのフェデレーションを使用して にアクセスすることを人間 AWS のユーザーに要求する](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集合を指定し、大量のユーザーに対するアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーに関するユースケース](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可を持つアイデンティティであり、一時的な認証情報を提供します。[ユーザーから IAM ロール \(コンソール\) に切り替えるか、または API オペレーションを呼び出す](#)

ことで、[ロール](#)を引き受けることができます。AWS CLI AWS 詳細については、「IAM ユーザーガイド」の「[ロールを引き受けるための各種方法](#)」を参照してください。

IAM ロールは、フェデレーションユーザーアクセス、一時的な IAM ユーザーのアクセス許可、クロスアカウントアクセス、クロスサービスアクセス、および Amazon EC2 で実行するアプリケーションに役立ちます。詳細については、IAM ユーザーガイドの [IAM でのクロスアカウントリソースアクセス](#) を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられている場合のアクセス許可を定義します。は、プリンシパルがリクエストを行うときにこれらのポリシー AWS を評価します。ほとんどのポリシーは JSON ドキュメント AWS としてに保存されます。JSON ポリシードキュメントの詳細については、「IAM ユーザーガイド」の「[JSON ポリシー概要](#)」を参照してください。

管理者は、ポリシーを使用して、どのプリンシパルがどのリソースに対して、どのような条件でアクションを実行できるかを定義することで、誰が何にアクセスできるかを指定します。

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成してロールに追加し、このロールをユーザーが引き受けられるようにします。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

アイデンティティベースのポリシーは、アイデンティティ (ユーザー、グループ、またはロール) にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、アイデンティティがどのリソースに対してどのような条件下でどのようなアクションを実行できるかを制御します。アイデンティティベースポリシーの作成方法については、IAM ユーザーガイドの [カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#) を参照してください。

アイデンティティベースのポリシーは、インラインポリシー (単一の ID に直接埋め込む) または管理ポリシー (複数の ID にアタッチされたスタンドアロンポリシー) にすることができます。管理ポリシーとインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「[管理ポリシーとインラインポリシーのいずれかを選択する](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。例としては、IAM ロール信頼ポリシーや Amazon S3 バケットポリシーなどがあります。リソースベースのポ

リシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

その他のポリシータイプ

AWS は、より一般的なポリシータイプによって付与されるアクセス許可の上限を設定できる追加のポリシータイプをサポートしています。

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の「[IAM エンティティのアクセス許可境界](#)」を参照してください。
- サービスコントロールポリシー (SCP) - AWS Organizations内の組織または組織単位の最大のアクセス許可を指定します。詳細については、「AWS Organizations ユーザーガイド」の「[サービスコントロールポリシー](#)」を参照してください。
- リソースコントロールポリシー (RCP) – は、アカウント内のリソースで利用できる最大数のアクセス許可を定義します。詳細については、「AWS Organizations ユーザーガイド」の「[リソースコントロールポリシー \(RCP\)](#)」を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の「[セッションポリシー](#)」を参照してください。

複数のポリシータイプ

1つのリクエストに複数のタイプのポリシーが適用されると、結果として作成されるアクセス許可を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「[ポリシー評価ロジック](#)」を参照してください。

Deadline Cloud と IAM の連携方法

IAM を使用して Deadline Cloud へのアクセスを管理する前に、Deadline Cloud で使用できる IAM 機能について説明します。

AWS Deadline Cloud で使用できる IAM 機能

IAM 機能	Deadline Cloud のサポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	あり
転送アクセスセッション (FAS)	あり
サービスロール	あり
サービスリンクロール	いいえ

Deadline Cloud およびその他の [がほとんどの IAM 機能と AWS のサービス連携する方法の概要](#)については、IAM ユーザーガイドの[AWS 「IAM と連携する のサービス」](#)を参照してください。

Deadline Cloud のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の[「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」](#)を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。JSON ポリシーで使用できるすべての要素に

ついて学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

Deadline Cloud のアイデンティティベースのポリシーの例

Deadline Cloud アイデンティティベースのポリシーの例を表示するには、「」を参照してください [Deadline Cloud のアイデンティティベースのポリシーの例](#)。

Deadline Cloud 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシー や Amazon S3 バケットポリシー があげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーで、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、またはを含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM エンティティを、リソースベースのポリシーのプリンシパルとして指定します。詳細については、IAM ユーザーガイドの [IAM でのクロスアカウントリソースアクセス](#) を参照してください。

Deadline Cloud のポリシーアクション

ポリシーアクションのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの Action 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

Deadline Cloud アクションのリストを確認するには、「サービス認可リファレンス」の「[Deadline Cloud AWS で定義されるアクション](#)」を参照してください。

Deadline Cloud のポリシーアクションは、アクションの前に次のプレフィックスを使用します。

```
deadline
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [  
  "deadline:action1",  
  "deadline:action2"  
]
```

Deadline Cloud アイデンティティベースのポリシーの例を表示するには、「」を参照してください [Deadline Cloud のアイデンティティベースのポリシーの例](#)。

Deadline Cloud のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ベストプラクティスとして、[Amazon リソースネーム \(ARN\)](#) を使用してリソースを指定します。リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

Deadline Cloud リソースタイプとその ARNs 「[Deadline Cloud AWS で定義されるリソース](#)」を参照してください。各リソースの ARN を指定できるアクションについては、「[Deadline Cloud AWS で定義されるアクション](#)」を参照してください。

Deadline Cloud アイデンティティベースのポリシーの例を表示するには、「」を参照してください [Deadline Cloud のアイデンティティベースのポリシーの例](#)。

Deadline Cloud のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。イコールや未満などの[条件演算子](#)を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

Deadline Cloud 条件キーのリストを確認するには、「サービス認可リファレンス」の[AWS 「Deadline Cloud の条件キー」](#)を参照してください。条件キーを使用できるアクションとリソースについては、「[Deadline Cloud AWS で定義されるアクション](#)」を参照してください。

Deadline Cloud アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Deadline Cloud のアイデンティティベースのポリシーの例](#)。

Deadline Cloud ACLs

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Deadline Cloud での ABAC

ABAC (ポリシー内のタグ) のサポート: あり

属性ベースのアクセス制御 (ABAC) は、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可する ABAC ポリシーを設計できます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの[条件要素](#)でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC 認可でアクセス許可を定義する](#)」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM

ユーザーガイド」の「[属性ベースのアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

Deadline Cloud での一時的な認証情報の使用

一時的な認証情報のサポート: あり

一時的な認証情報は、AWS リソースへの短期的なアクセスを提供し、フェデレーションまたはスイッチロールの使用時に自動的に作成されます。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM ユーザーガイド」の「[IAM の一時的な認証情報](#)」および「[AWS のサービスと IAM との連携](#)」を参照してください。

Deadline Cloud の転送アクセスセッション

転送アクセスセッション (FAS) のサポート: あり

転送アクセスセッション (FAS) は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする を使用します。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

Deadline Cloud のサービスロール

サービスロールのサポート: あり

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの [AWS のサービスに許可を委任するロールを作成する](#) を参照してください。

Warning

サービスロールのアクセス許可を変更すると、Deadline Cloud の機能が破損する可能性があります。Deadline Cloud が指示する場合にのみ、サービスロールを編集します。

Deadline Cloud のサービスにリンクされたロール

サービスにリンクされたロールのサポート: なし

サービスにリンクされたロールは、にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールはに表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「[IAM と提携するAWS のサービス](#)」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つけます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Deadline Cloud のアイデンティティベースのポリシーの例

デフォルトでは、ユーザーとロールには Deadline Cloud リソースを作成または変更するアクセス許可はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。

これらのサンプルの JSON ポリシードキュメントを使用して IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「[IAM ポリシーを作成する \(コンソール\)](#)」を参照してください。

各リソースタイプの ARNs [AWS 「Deadline Cloud のアクション、リソース、および条件キー」](#) を参照してください。

トピック

- [ポリシーに関するベストプラクティス](#)
- [Deadline Cloud コンソールの使用](#)
- [コンソールにアクセスするためのポリシー](#)
- [キューにジョブを送信するポリシー](#)
- [ライセンスエンドポイントの作成を許可するポリシー](#)
- [特定のファームキューのモニタリングを許可するポリシー](#)

ポリシーに関するベストプラクティス

ID ベースのポリシーは、アカウント内の Deadline Cloud リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションでは、AWS アカウントに費用が発生する場合があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらはで使用できます AWS アカウント。ユースケースに固有のAWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、IAM ユーザーガイドの [AWS マネージドポリシー](#) または [ジョブ機能のAWS マネージドポリシー](#) を参照してください。
- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイドの [IAM でのポリシーとアクセス許可](#) を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションがなどの特定のを通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイドの [IAM JSON ポリシー要素:条件](#) を参照してください。
- IAM アクセスアナライザー を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM アクセスアナライザー は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイドの [IAM Access Analyzer でポリシーを検証する](#) を参照してください。
- 多要素認証 (MFA) を要求する – IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイドの [MFA を使用した安全な API アクセス](#) を参照してください。

IAM でのベストプラクティスの詳細については、IAM ユーザーガイドの [IAM でのセキュリティのベストプラクティス](#) を参照してください。

Deadline Cloud コンソールの使用

AWS Deadline Cloud コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、の Deadline Cloud リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成

すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみ呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが引き続き Deadline Cloud コンソールを使用できるようにするには、エンティティに Deadline Cloud *ConsoleAccess* または *ReadOnly* AWS マネージドポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「[ユーザーへのアクセス許可の追加](#)」を参照してください。

コンソールにアクセスするためのポリシー

Deadline Cloud コンソールのすべての機能へのアクセスを許可するには、フルアクセスを付与するユーザーまたはロールにこの ID ポリシーをアタッチします。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EC2InstanceTypeSelection",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstanceTypeOfferings",
        "ec2:DescribeInstanceTypes",
        "ec2:GetInstanceTypesFromInstanceRequirements",
        "pricing:GetProducts"
      ],
      "Resource": ["*"]
    },
    {
      "Sid": "VPCResourceSelection",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups"
      ],
      "Resource": ["*"]
    }
  ],
}
```

```
{
  "Sid": "ViewVpcLatticeResources",
  "Effect": "Allow",
  "Action": [
    "vpc-lattice:ListResourceConfigurations",
    "vpc-lattice:GetResourceConfiguration",
    "vpc-lattice:GetResourceGateway"
  ],
  "Resource": ["*"]
},
{
  "Sid": "ManageVpcEndpointsViaDeadline",
  "Effect": "Allow",
  "Action": [
    "ec2:CreateVpcEndpoint",
    "ec2:DescribeVpcEndpoints",
    "ec2>DeleteVpcEndpoints",
    "ec2:CreateTags"
  ],
  "Resource": ["*"],
  "Condition": {
    "StringEquals": { "aws:CalledViaFirst": "deadline.amazonaws.com" }
  }
},
{
  "Sid": "ChooseJobAttachmentsBucket",
  "Effect": "Allow",
  "Action": ["s3:GetBucketLocation", "s3:ListAllMyBuckets"],
  "Resource": "*"
},
{
  "Sid": "CreateDeadlineCloudLogGroups",
  "Effect": "Allow",
  "Action": ["logs:CreateLogGroup"],
  "Resource": "arn:aws:logs:*:*:log-group:/aws/deadline/*",
  "Condition": {
    "StringLike": { "aws:CalledViaFirst": "deadline.amazonaws.com" }
  }
},
{
  "Sid": "ValidateDependencies",
  "Effect": "Allow",
  "Action": ["s3:ListBucket"],
  "Resource": "*",
```

```
    "Condition": {
      "StringLike": { "aws:CalledViaFirst": "deadline.amazonaws.com" }
    },
    {
      "Sid": "RoleSelection",
      "Effect": "Allow",
      "Action": ["iam:GetRole", "iam:ListRoles",
"iam:ListAttachedRolePolicies"],
      "Resource": "*"
    },
    {
      "Sid": "PassRoleToDeadlineCloud",
      "Effect": "Allow",
      "Action": ["iam:PassRole"],
      "Condition": {
        "StringLike": { "iam:PassedToService": "deadline.amazonaws.com" }
      },
      "Resource": "*"
    },
    {
      "Sid": "KMSKeySelection",
      "Effect": "Allow",
      "Action": ["kms:ListKeys", "kms:ListAliases"],
      "Resource": "*"
    },
    {
      "Sid": "IdentityStoreReadOnly",
      "Effect": "Allow",
      "Action": [
        "identitystore:DescribeUser",
        "identitystore:DescribeGroup",
        "identitystore:ListGroups",
        "identitystore:ListUsers",
        "identitystore:IsMemberInGroups",
        "identitystore:ListGroupMemberships",
        "identitystore:ListGroupMembershipsForMember",
        "identitystore:GetGroupMembershipId"
      ],
      "Resource": "*"
    },
    {
      "Sid": "OrganizationAndIdentityCenterIdentification",
      "Effect": "Allow",
```

```
"Action": [
  "sso:ListDirectoryAssociations",
  "organizations:DescribeAccount",
  "organizations:DescribeOrganization",
  "sso:DescribeRegisteredRegions",
  "sso:GetManagedApplicationInstance",
  "sso:GetSharedSsoConfiguration",
  "sso:ListInstances",
  "sso:GetApplicationAssignmentConfiguration",
  "sso:GetSSOStatus",
  "sso:ListRegions",
  "sso:DescribeRegion"
],
"Resource": "*"
},
{
  "Sid": "ManagedDeadlineCloudIDCAApplication",
  "Effect": "Allow",
  "Action": [
    "sso:CreateApplication",
    "sso:PutApplicationAssignmentConfiguration",
    "sso:PutApplicationAuthenticationMethod",
    "sso:PutApplicationGrant",
    "sso>DeleteApplication",
    "sso:UpdateApplication"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": { "aws:CalledViaFirst": "deadline.amazonaws.com" }
  }
},
{
  "Sid": "ChooseSecret",
  "Effect": "Allow",
  "Action": ["secretsmanager:ListSecrets"],
  "Resource": "*"
},
{
  "Sid": "DeadlineMembershipActions",
  "Effect": "Allow",
  "Action": [
    "deadline:AssociateMemberToFarm",
    "deadline:AssociateMemberToFleet",
    "deadline:AssociateMemberToQueue",
```

```
        "deadline:AssociateMemberToJob",
        "deadline:DisassociateMemberFromFarm",
        "deadline:DisassociateMemberFromFleet",
        "deadline:DisassociateMemberFromQueue",
        "deadline:DisassociateMemberFromJob",
        "deadline:ListFarmMembers",
        "deadline:ListFleetMembers",
        "deadline:ListQueueMembers",
        "deadline:ListJobMembers"
    ],
    "Resource": ["*"]
},
{
    "Sid": "DeadlineControlPlaneActions",
    "Effect": "Allow",
    "Action": [
        "deadline:CreateMonitor",
        "deadline:GetMonitor",
        "deadline:UpdateMonitor",
        "deadline>DeleteMonitor",
        "deadline:ListMonitors",
        "deadline:CreateFarm",
        "deadline:GetFarm",
        "deadline:UpdateFarm",
        "deadline>DeleteFarm",
        "deadline:ListFarms",
        "deadline:CreateQueue",
        "deadline:GetQueue",
        "deadline:UpdateQueue",
        "deadline>DeleteQueue",
        "deadline:ListQueues",
        "deadline:CreateFleet",
        "deadline:GetFleet",
        "deadline:UpdateFleet",
        "deadline>DeleteFleet",
        "deadline:ListFleets",
        "deadline:ListWorkers",
        "deadline:CreateQueueFleetAssociation",
        "deadline:GetQueueFleetAssociation",
        "deadline:UpdateQueueFleetAssociation",
        "deadline>DeleteQueueFleetAssociation",
        "deadline:ListQueueFleetAssociations",
        "deadline:CreateQueueEnvironment",
        "deadline:GetQueueEnvironment",
```

```
    "deadline:UpdateQueueEnvironment",
    "deadline>DeleteQueueEnvironment",
    "deadline>ListQueueEnvironments",
    "deadline>CreateLimit",
    "deadline:GetLimit",
    "deadline:UpdateLimit",
    "deadline>DeleteLimit",
    "deadline>ListLimits",
    "deadline>CreateQueueLimitAssociation",
    "deadline:GetQueueLimitAssociation",
    "deadline>DeleteQueueLimitAssociation",
    "deadline:UpdateQueueLimitAssociation",
    "deadline>ListQueueLimitAssociations",
    "deadline>CreateStorageProfile",
    "deadline:GetStorageProfile",
    "deadline:UpdateStorageProfile",
    "deadline>DeleteStorageProfile",
    "deadline>ListStorageProfiles",
    "deadline>ListStorageProfilesForQueue",
    "deadline>ListBudgets",
    "deadline:TagResource",
    "deadline:UntagResource",
    "deadline>ListTagsForResource",
    "deadline>CreateLicenseEndpoint",
    "deadline:GetLicenseEndpoint",
    "deadline>DeleteLicenseEndpoint",
    "deadline>ListLicenseEndpoints",
    "deadline>ListAvailableMeteredProducts",
    "deadline>ListMeteredProducts",
    "deadline:PutMeteredProduct",
    "deadline>DeleteMeteredProduct",
    "deadline:GetMonitorSettings",
    "deadline:UpdateMonitorSettings",
    "deadline:GetVolume",
    "deadline>ListVolumes",
    "deadline>DeleteVolume"
  ],
  "Resource": ["*"]
}]
}
```

キューにジョブを送信するポリシー

この例では、特定のファーム内の特定のキューにジョブを送信するアクセス許可を付与するスコープダウンポリシーを作成します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SubmitJobsFarmAndQueue",
      "Effect": "Allow",
      "Action": "deadline:CreateJob",
      "Resource": "arn:aws:deadline:us-east-1:111122223333:farm/FARM_A/queue/QUEUE_B/job/*"
    }
  ]
}
```

ライセンスエンドポイントの作成を許可するポリシー

この例では、ライセンスエンドポイントを作成および管理するために必要なアクセス許可を付与するスコープダウンポリシーを作成します。このポリシーを使用して、ファームに関連付けられた VPC のライセンスエンドポイントを作成します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "CreateLicenseEndpoint",
    "Effect": "Allow",
    "Action": [
      "deadline:CreateLicenseEndpoint",
      "deadline>DeleteLicenseEndpoint",
      "deadline:GetLicenseEndpoint",
      "deadline>ListLicenseEndpoints",
      "deadline:PutMeteredProduct",
      "deadline>DeleteMeteredProduct",
    ]
  }]
}
```

```
        "deadline:ListMeteredProducts",
        "deadline:ListAvailableMeteredProducts",
        "ec2:CreateVpcEndpoint",
        "ec2:DescribeVpcEndpoints",
        "ec2>DeleteVpcEndpoints"
    ],
    "Resource": [
        "arn:aws:deadline:*:111122223333:*",
        "arn:aws:ec2:*:111122223333:vpc-endpoint/*"
    ]
}]]
}
```

特定のファームキューのモニタリングを許可するポリシー

この例では、特定のファームの特定のキュー内のジョブをモニタリングするアクセス許可を付与するスコープダウンポリシーを作成します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "MonitorJobsFarmAndQueue",
    "Effect": "Allow",
    "Action": [
      "deadline:SearchJobs",
      "deadline:ListJobs",
      "deadline:GetJob",
      "deadline:SearchSteps",
      "deadline:ListSteps",
      "deadline:ListStepConsumers",
      "deadline:ListStepDependencies",
      "deadline:GetStep",
      "deadline:SearchTasks",
      "deadline:ListTasks",
      "deadline:GetTask",
      "deadline:ListSessions",
      "deadline:GetSession",
      "deadline:ListSessionActions",
      "deadline:GetSessionAction"
    ]
  }]
}
```

```
    ],
    "Resource": [
      "arn:aws:deadline:us-east-1:123456789012:farm/FARM_A/queue/QUEUE_B",
      "arn:aws:deadline:us-east-1:123456789012:farm/FARM_A/queue/QUEUE_B/*"
    ]
  }
}
```

AWS Deadline Cloud の マネージドポリシー

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンポリシーです AWS。AWS 管理ポリシーは、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できるように、多くの一般的なユースケースにアクセス許可を提供するように設計されています。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケースに固有の[カスタマー管理ポリシー](#)を定義して、アクセス許可を絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS マネージドポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。AWS は、新しい が起動されるか、新しい API オペレーション AWS のサービス が既存のサービスで使用できるようになったときに、AWS マネージドポリシーを更新する可能性が高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

AWS 管理ポリシー: AWSDeadlineCloud-FleetWorker

AWSDeadlineCloud-FleetWorker ポリシーを (IAM) ID に AWS Identity and Access Management アタッチできます。

このポリシーは、このフリートのワーカーに、サービスへの接続とサービスからのタスクの受信に必要なアクセス許可を付与します。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `deadline` – プリンシパルがフリート内のワーカーを管理できるようにします。

ポリシーの詳細の JSON リストについては、[AWSDeadlineCloud-FleetWorker](#)」を参照してください。

AWS 管理ポリシー: AWSDeadlineCloud-WorkerHost

AWSDeadlineCloud-WorkerHost ポリシーを IAM アイデンティティにアタッチできます。

このポリシーは、最初に サービスに接続するために必要なアクセス許可を付与します。Amazon Elastic Compute Cloud (Amazon EC2) インスタンスプロファイルとして使用できます。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `deadline` – ワーカーの作成、ワーカーのフリートルールの引き受け、ワーカーへのタグの適用をユーザーに許可します

ポリシーの詳細の JSON リストについては、[AWSDeadlineCloud-WorkerHost](#)」を参照してください。

AWS 管理ポリシー: AWSDeadlineCloud-UserAccessFarms

AWSDeadlineCloud-UserAccessFarms ポリシーを IAM アイデンティティにアタッチできます。

このポリシーにより、ユーザーは自分がメンバーであるファームとそのメンバーシップレベルに基づいてファームデータにアクセスできます。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `deadline` – ファームデータへのアクセスをユーザーに許可します。
- `ec2` – ユーザーが Amazon EC2 インスタンスタイプの詳細を表示できるようにします。
- `identitystore` – ユーザーがユーザー名とグループ名を表示できるようにします。

- kms – ユーザーが AWS Key Management Service (IAM Identity Center AWS KMS) インスタンスの AWS IAM アイデンティティセンター () カスタマーマネージドキーを設定できるようにします。

ポリシーの詳細の JSON リストについては、[AWSDeadlineCloud-UserAccessFarms](#)」を参照してください。

AWS マネージドポリシー: AWSDeadlineCloud-UserAccessFleets

AWSDeadlineCloud-UserAccessFleets ポリシーを IAM アイデンティティにアタッチできます。

このポリシーにより、ユーザーは自分がメンバーであるファームとそのメンバーシップレベルに基づいてフリートデータにアクセスできます。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- deadline – ファームデータへのアクセスをユーザーに許可します。
- ec2 – ユーザーが Amazon EC2 インスタンスタイプの詳細を表示できるようにします。
- identitystore – ユーザーがユーザー名とグループ名を表示できるようにします。

ポリシーの詳細の JSON リストについては、[AWS AWSDeadlineCloud-UserAccessFleets](#)」を参照してください。

AWS 管理ポリシー: AWSDeadlineCloud-UserAccessJobs

AWSDeadlineCloud-UserAccessJobs ポリシーを IAM アイデンティティにアタッチできます。

このポリシーにより、ユーザーは自分がメンバーであるファームとそのメンバーシップレベルに基づいてジョブデータにアクセスできます。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- deadline – ファームデータへのアクセスをユーザーに許可します。
- ec2 – ユーザーが Amazon EC2 インスタンスタイプの詳細を表示できるようにします。

- `identitystore` – ユーザーがユーザー名とグループ名を表示できるようにします。

ポリシーの詳細の JSON リストについては、[AWS AWSDeadlineCloud-UserAccessJobs](#)」を参照してください。

AWS 管理ポリシー: AWSDeadlineCloud-UserAccessQueues

AWSDeadlineCloud-UserAccessQueues ポリシーを IAM アイデンティティにアタッチできます。

このポリシーにより、ユーザーは自分がメンバーであるファームとそのメンバーシップレベルに基づいてキューデータにアクセスできます。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `deadline` – ファームデータへのアクセスをユーザーに許可します。
- `ec2` – ユーザーが Amazon EC2 インスタンスタイプの詳細を表示できるようにします。
- `identitystore` – ユーザーがユーザー名とグループ名を表示できるようにします。

ポリシーの詳細の JSON リストについては、[AWSDeadlineCloud-UserAccessQueues](#)」を参照してください。

AWS マネージドポリシーの Deadline Cloud 更新

このサービスがこれらの変更の追跡を開始してからの Deadline Cloud の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動アラートについては、Deadline Cloud Document 履歴ページの RSS フィードにサブスクライブしてください。

変更	説明	日付
AWSDeadlineCloud-UserAccessFarms – 変更	Deadline Cloud に新しいアクションが追加され <code>kms:Decrypt</code> 、IAM Identity Center イン	2025 年 12 月 22 日

変更	説明	日付
	スタンスで AWS KMS カスタマーマネージドキーを使用できるようになりました。	
AWSDeadlineCloud-WorkerHost – 変更	Deadline Cloud は <code>deadline: TagResource</code> 、フリート内のワーカーに関連付けられたタグを追加および表示 <code>deadline: ListTagsForResource</code> できるように、新しいアクション およびを追加しました。	2025 年 5 月 30 日
AWSDeadlineCloud-UserAccessFarms – 変更 AWSDeadlineCloud-UserAccessJobs – 変更 AWSDeadlineCloud-UserAccessQueues – 変更	Deadline Cloud に新しいアクション <code>deadline: GetJobTemplate</code> と <code>deadline: ListJobParameterDefinitions</code> 、ジョブを再送信できるようになりました。	2024 年 10 月 7 日
Deadline Cloud が変更の追跡を開始しました	Deadline Cloud は、AWS マネージドポリシーの変更の追跡を開始しました。	2024 年 4 月 2 日

サービスロール

Deadline Cloud が IAM サービスロールを使用する方法

Deadline Cloud は自動的に IAM ロールを引き受け、ワーカー、ジョブ、および Deadline Cloud モニターに一時的な認証情報を提供します。このアプローチにより、ロールベースのアクセスコントロールを通じてセキュリティを維持しながら、手動の認証情報管理が不要になります。

モニター、フリート、キューを作成するときは、Deadline Cloud がユーザーに代わって引き受ける IAM ロールを指定します。その後、ワーカーと Deadline Cloud モニターは、アクセスするためにこれらのロールから一時的な認証情報を受け取ります AWS のサービス。

フリートロール

Deadline Cloud ワーカーが作業を受け取り、その作業の進行状況をレポートするために必要なアクセス許可を付与するようにフリートロールを設定します。

通常、このロールを自分で設定する必要はありません。このロールは、Deadline Cloud コンソールで作成して、必要なアクセス許可を含めることができます。トラブルシューティングのためにこのロールの詳細を理解するには、次のガイドを使用します。

プログラムでフリートを作成または更新する場合は、CreateFleet または UpdateFleet API オペレーションを使用してフリートロール ARN を指定します。

フリートロールの動作

フリートロールは、ワーカーに次のアクセス許可を付与します。

- 新しい作業を受け取り、進行中の作業の進捗状況を Deadline Cloud サービスに報告する
- ワーカーのライフサイクルとステータスを管理する
- ワーカーログのログイベントを Amazon CloudWatch Logs に記録する

フリートロールの信頼ポリシーを設定する

フリートロールは Deadline Cloud サービスを信頼し、特定のファームに限定する必要があります。

ベストプラクティスとして、信頼ポリシーには混乱した代理保護のセキュリティ条件を含める必要があります。混乱した代理の保護の詳細については、「Deadline Cloud ユーザーガイド」の[「混乱した代理」](#)を参照してください。

- `aws:SourceAccount` は、同じのリソースのみがこのロールを引き受け AWS アカウント することができますようにします。
- `aws:SourceArn` は、ロールの引き受けを特定の Deadline Cloud ファームに制限します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDeadlineCredentialsService",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
```

```
    "Principal": {
      "Service": "credentials.deadline.amazonaws.com"
    },
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "YOUR_ACCOUNT_ID"
      },
      "ArnEquals": {
        "aws:SourceArn": "arn:aws:deadline:REGION:YOUR_ACCOUNT_ID:farm/YOUR_FARM_ID"
      }
    }
  }
}
```

フリートロールのアクセス許可をアタッチする

フリートロールに次の AWS 管理ポリシーをアタッチします。

[AWSDeadlineCloud-FleetWorker](#)

この管理ポリシーは、以下のアクセス許可を提供します。

- `deadline:AssumeFleetRoleForWorker` - ワーカーが認証情報を更新できるようにします。
- `deadline:UpdateWorker` - ワーカーがステータスを更新できるようにします (終了時に `STOPPED` など)。
- `deadline:UpdateWorkerSchedule` - 作業の取得と進行状況の報告用。
- `deadline:BatchGetJobEntity` - ジョブ情報を取得する場合。
- `deadline:AssumeQueueRoleForWorker` - ジョブの実行中にキューロールの認証情報にアクセスする場合。

暗号化されたファームに KMS アクセス許可を追加する

ファームが KMS キーを使用して作成された場合は、これらのアクセス許可をフリートロールに追加して、ワーカーがファーム内の暗号化されたデータにアクセスできるようにします。

KMS アクセス許可は、ファームに KMS キーが関連付けられている場合にのみ必要です。 `kms:ViaService` 条件は形式を使用する必要があります `deadline.{region}.amazonaws.com`。

フリートを作成すると、そのフリートの CloudWatch Logs ロググループが作成されます。ワーカーのアクセス許可は、Deadline Cloud サービスによって使用され、その特定のワーカー専用のログストリームを作成します。ワーカーをセットアップして実行すると、ワーカーはこれらのアクセス許可を使用してログイベントを CloudWatch Logs に直接送信します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateLogStream",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream"
      ],
      "Resource": "arn:aws:logs:REGION:YOUR_ACCOUNT_ID:log-group:/aws/
deadline/YOUR_FARM_ID/*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": [
            "deadline.REGION.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "ManageLogEvents",
      "Effect": "Allow",
      "Action": [
        "logs:PutLogEvents",
        "logs:GetLogEvents"
      ],
      "Resource": "arn:aws:logs:REGION:YOUR_ACCOUNT_ID:log-group:/aws/
deadline/YOUR_FARM_ID/*"
    },
    {
      "Sid": "ManageKmsKey",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:GenerateDataKey"
      ],
      "Resource": "YOUR_FARM_KMS_KEY_ARN",
```

```
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "deadline.REGION.amazonaws.com"
      }
    }
  ]
}
```

フリートロールの変更

フリートロールのアクセス許可はカスタマイズできません。説明されているアクセス許可は常に必要であり、追加のアクセス許可を追加しても効果はありません。

カスタマーマネージドフリートホストロール

Amazon EC2 インスタンスまたはオンプレミスホストでカスタマーマネージドフリートを使用する場合は、WorkerHost ロールを設定します。

WorkerHost ロールの動作

WorkerHost ロールは、カスタマーマネージドフリートホストでワーカーをブートストラップします。ホストが以下を行うために必要な最小限のアクセス許可を提供します。

- Deadline Cloud でワーカーを作成する
- フリートロールを引き受けて運用認証情報を取得する
- フリートタグを使用してワーカーにタグを付ける (タグ伝達が有効になっている場合)

WorkerHost ロールのアクセス許可を設定する

次の AWS 管理ポリシーを WorkerHost ロールにアタッチします。

[AWSDeadlineCloud-WorkerHost](#)

この管理ポリシーは、以下のアクセス許可を提供します。

- `deadline:CreateWorker` - ホストが新しいワーカーを登録できるようにします。
- `deadline:AssumeFleetRoleForWorker` - ホストがフリートロールを引き受けることを許可します。
- `deadline:TagResource` - 作成中のワーカーのタグ付けを許可します (有効になっている場合)。

- `deadline:ListTagsForResource` - 伝播用のフリートタグの読み取りを許可します。

ブートストラッププロセスを理解する

WorkerHost ロールは、ワーカーの初回起動時にのみ使用されます。

1. ワーカーエージェントは、WorkerHost 認証情報を使用してホストで起動します。
2. を呼び出し`deadline:CreateWorker`で Deadline Cloud に登録します。
3. 次に、 を呼び出し`deadline:AssumeFleetRoleForWorker`でフリートロールの認証情報を取得します。
4. この時点から、ワーカーはすべてのオペレーションにフリートロール認証情報のみを使用します。

WorkerHost ロールは、ワーカーの実行開始後は使用されません。このポリシーは、サービスマネージドフリートには必要ありません。サービスマネージドフリートでは、ブートストラップが自動的に実行されます。

キューロール

キューロールは、タスクを処理するときにワーカーによって引き受けられます。このロールは、タスクを完了するために必要なアクセス許可を提供します。

プログラムでキューを作成または更新する場合は、`CreateQueue` または `UpdateQueue` API オペレーションを使用してキューロール ARN を指定します。

キューロールの信頼ポリシーを設定する

キューロールは Deadline Cloud サービスを信頼する必要があります。

ベストプラクティスとして、信頼ポリシーには混乱した代理保護のセキュリティ条件を含める必要があります。混乱した代理の保護の詳細については、「Deadline Cloud ユーザーガイド」の「[混乱した代理](#)」を参照してください。

- `aws:SourceAccount` は、同じのリソースのみがこのロールを引き受け AWS アカウント ことができるようにします。
- `aws:SourceArn` は、ロールの引き受けを特定の Deadline Cloud フォームに制限します。

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "Service": [
        "credentials.deadline.amazonaws.com",
        "deadline.amazonaws.com"
      ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "YOUR_ACCOUNT_ID"
      },
      "ArnEquals": {
        "aws:SourceArn": "arn:aws:deadline:us-west-2:123456789012:farm/{farm-id}"
      }
    }
  }
]
```

キューロールのアクセス許可を理解する

キューロールは 1 つの管理ポリシーを使用しません。代わりに、コンソールでキューを設定すると、Deadline Cloud は設定に基づいてキューのカスタムポリシーを作成します。

この自動作成されたポリシーは、以下へのアクセスを提供します。

ジョブアタッチメント

ジョブの入出カファイル用に指定された Amazon S3 バケットへの読み取りおよび書き込みアクセス:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:PutObject",
    "s3:ListBucket",
    "s3:GetBucketLocation"
  ],
  "Resource": [
    "arn:aws:s3:::YOUR_JOB_ATTACHMENTS_BUCKET",
```

```
    "arn:aws:s3:::YOUR_JOB_ATTACHMENTS_BUCKET/YOUR_PREFIX/*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "YOUR_ACCOUNT_ID"
    }
  }
}
```

ジョブのログ

このキュー内のジョブの CloudWatch Logs への読み取りアクセス。各キューには独自のロググループがあり、各セッションには独自のログストリームがあります。

```
{
  "Effect": "Allow",
  "Action": [
    "logs:GetLogEvents"
  ],
  "Resource": "arn:aws:logs:REGION:YOUR_ACCOUNT_ID:log-group:/aws/
deadline/YOUR_FARM_ID/*"
}
```

サードパーティー製ソフトウェア

Deadline Cloud でサポートされているサードパーティーソフトウェア (Maya、Blender など) をダウンロードするアクセス:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:ListBucket",
    "s3:GetObject"
  ],
  "Resource": "*",
  "Condition": {
    "ArnLike": {
      "s3:DataAccessPointArn": "arn:aws:s3:*:*:accesspoint/deadline-software-*"
    },
    "StringEquals": {
      "s3:AccessPointNetworkOrigin": "VPC"
    }
  }
}
```

```
}
```

ジョブのアクセス許可を追加する

ジョブ AWS のサービス がアクセスする必要がある のキューロールにアクセス許可を追加します。OpenJobDescription ステップスクリプトを記述すると、AWS CLI と SDK はキューロールの認証情報を自動的に使用します。これを使用して、ジョブの完了に必要な追加サービスにアクセスします。

ユースケースの例を以下に示します。

- カスタムデータを取得するための
- カスタムライセンスサーバーにトンネルする SSM アクセス許可
- カスタムメトリクスを発行するための CloudWatch
- 動的ワークフロー用の新しいジョブを作成する Deadline Cloud アクセス許可

キューロールの認証情報の使用方法

Deadline Cloud はキューロールの認証情報を以下に提供します。

- ジョブ実行中のワーカー
- Deadline Cloud CLI を介したユーザーと、ジョブの添付ファイルやログを操作する際のモニタリング

Deadline Cloud は、キューごとに個別の CloudWatch Logs ロググループを作成します。ジョブはキューロールの認証情報を使用して、キューのロググループにログを書き込みます。Deadline Cloud CLI とモニターは、キューロール (経由deadline:AssumeQueueRoleForRead) を使用してキューのロググループからジョブログを読み込みます。Deadline Cloud CLI とモニターは、キューロール (経由deadline:AssumeQueueRoleForUser) を使用してジョブアタッチメントデータをアップロードまたはダウンロードします。

ロールをモニタリングする

Deadline Cloud モニターのウェブアプリケーションとデスクトップアプリケーションに Deadline Cloud リソースへのアクセスを許可するようにモニターロールを設定します。

プログラムでモニターを作成または更新する場合は、CreateMonitor または UpdateMonitor API オペレーションを使用してモニターロール ARN を指定します。

モニターロールの動作

モニターロールにより、Deadline Cloud モニターはエンドユーザーに以下へのアクセスを提供できます。

- Deadline Cloud Integrated Submitters、CLI、モニターに必要な基本機能
- エンドユーザー向けのカスタム機能

モニターロールの信頼ポリシーを設定する

モニターロールは Deadline Cloud サービスを信頼する必要があります。

ベストプラクティスとして、信頼ポリシーには混乱した代理保護のセキュリティ条件を含める必要があります。混乱した代理の保護の詳細については、「Deadline Cloud ユーザーガイド」の「[混乱した代理](#)」を参照してください。

`aws:SourceAccount` は、同じのリソースのみがこのロールを引き受け AWS アカウント ことができるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "credentials.deadline.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "YOUR_ACCOUNT_ID"
        }
      }
    }
  ]
}
```

モニターロールのアクセス許可をアタッチする

基本オペレーションのために、以下のすべての AWS 管理ポリシーをモニターロールにアタッチします。

- [AWSDeadlineCloud-UserAccessFarms](#)
- [AWSDeadlineCloud-UserAccessFleets](#)
- [AWSDeadlineCloud-UserAccessJobs](#)
- [AWSDeadlineCloud-UserAccessQueues](#)

モニターロールの仕組み

Deadline Cloud モニターを使用する場合、サービスユーザーは AWS IAM アイデンティティセンター (IAM Identity Center) を使用してサインインし、モニターロールが引き受けられます。引き受けたロール認証情報は、ファーム、フリート、キュー、その他の情報のリストなど、モニター UI を表示するためにモニターアプリケーションによって使用されます。

Deadline Cloud モニターデスクトップアプリケーションを使用する場合、これらの認証情報は、エンドユーザーから提供されたプロファイル名に対応する名前付き AWS 認証情報プロファイルを使用してワークステーションで利用可能になります。名前付きプロファイルの詳細については、[AWS SDK およびツールリファレンスガイド](#)を参照してください。

この名前付きプロファイルは、Deadline CLI と送信者が Deadline Cloud リソースにアクセスする方法です。

高度なユースケースに合わせてモニターロールをカスタマイズする

モニターロールをカスタマイズして、各アクセスレベル (ビューワー、コントリビューター、マネージャー、所有者) でユーザーが実行できる操作を変更したり、高度なワークフローのアクセス許可を追加したりできます。

アクセスレベルのアクセス許可のカスタマイズ

モニターロールにアタッチされた 4 つの AWS 管理ポリシーは、各アクセスレベルが実行できる操作を制御します。モニターロールにカスタムポリシーを追加して、`deadline:MembershipLevel` 条件キーを使用して特定のアクセスレベルのアクセス許可を付与または制限できます。

たとえば、コントリビューターがジョブ (通常はマネージャーと所有者に制限されます) を更新およびキャンセルできるようにするには、次のようなポリシーを追加します。

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
{
  "Effect": "Allow",
  "Action": "deadline:UpdateJob",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "deadline:MembershipLevel": "CONTRIBUTOR"
    }
  }
}
```

このポリシーを使用すると、寄稿者はジョブを送信することに加えて、ジョブを更新およびキャンセルできます。

高度なワークフローに対するアクセス許可の追加

カスタム IAM ポリシーをモニターロールに追加して、すべてのモニターユーザーに追加のアクセス許可を付与できます。これは、ユーザーが標準の Deadline Cloud 機能 AWS のサービス を超えてにアクセスする必要がある高度なスクリプトワークフローに役立ちます。

モニターロールを変更するときは、次のガイドラインに従ってください。

- 管理ポリシーを削除しないでください。これらのポリシーを削除すると、モニター機能が破損します。

Deadline Cloud Monitor がモニターロールの認証情報を使用する方法

Deadline Cloud Monitor は、認証時にモニターロールの認証情報を自動的に取得します。この機能を使用すると、デスクトップアプリケーションは標準のウェブブラウザで利用できる以上の拡張モニタリング機能を提供できます。

Deadline Cloud Monitor でログインすると、AWS CLI または他の AWS ツールで使用できるプロファイルが自動的に作成されます。このプロファイルでは、モニターロールの認証情報を使用して、モニターロールのアクセス許可 AWS のサービス に基づいて へのプログラムによるアクセスを許可します。

Deadline Cloud 送信者は同じように動作します。Deadline Cloud モニターによって作成されたプロファイルを使用して、適切なロールのアクセス許可 AWS のサービス でにアクセスします。

Deadline Cloud ロールの高度なカスタマイズ

Deadline Cloud ロールを追加のアクセス許可で拡張して、基本的なレンダリングワークフロー以外の高度なユースケースを実現できます。このアプローチでは、Deadline Cloud のアクセス管理システムを活用して、キューメンバーシップ AWS のサービスに基づいて追加のへのアクセスを制御します。

とのチームコラボレーション AWS CodeCommit

キューロールにアクセス AWS CodeCommit 許可を追加して、プロジェクトリポジトリでチームコラボレーションを有効にします。このアプローチでは、追加のユースケースに Deadline Cloud のアクセス管理システムを使用します。特定のキューにアクセスできるユーザーのみがこれらのアクセス AWS CodeCommit 許可を受け取るため、Deadline Cloud キューメンバーシップを通じてプロジェクトごとのリポジトリアクセスを管理できます。

これは、アーティストがレンダリングワークフローの一部として AWS CodeCommit リポジトリに保存されているプロジェクト固有のアセット、スクリプト、または設定ファイルにアクセスする必要があるシナリオに役立ちます。

キューロールにアクセス AWS CodeCommit 許可を追加する

アクセスを有効にするには、キューロールに次の AWS CodeCommit アクセス許可を追加します。

```
{
  "Effect": "Allow",
  "Action": [
    "codecommit:GitPull",
    "codecommit:GitPush",
    "codecommit:GetRepository",
    "codecommit:ListRepositories"
  ],
  "Resource": "arn:aws:codecommit:REGION:YOUR_ACCOUNT_ID:PROJECT_REPOSITORY"
}
```

アーティストワークステーションで認証情報プロバイダーを設定する

AWS CodeCommit アクセスに Deadline Cloud キュー認証情報を使用するように各アーティストワークステーションを設定します。この設定はワークステーションごとに 1 回行われます。

認証情報プロバイダーを設定するには

1. 認証情報プロバイダープロファイルを設定ファイル (~/.aws/config) AWS に追加します。

るユーザーのみが関連付けられたリポジトリにアクセスできるため、Deadline Cloud のキューメンバシップシステムを通じてきめ細かなアクセスコントロールが可能になります。

AWS Deadline Cloud のアイデンティティとアクセスのトラブルシューティング

以下の情報は、Deadline Cloud と IAM の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちます。

トピック

- [Deadline Cloud でアクションを実行する権限がありません](#)
- [iam:PassRole を実行する権限がありません](#)
- [自分の 以外のユーザーに Deadline Cloud リソース AWS アカウント へのアクセスを許可したい](#)

Deadline Cloud でアクションを実行する権限がありません

アクションを実行する権限がないというエラーが表示された場合は、そのアクションを実行できるようにポリシーを更新する必要があります。

次のエラー例は、mateojackson IAM ユーザーがコンソールを使用して、ある *my-example-widget* リソースに関する詳細情報を表示しようとしたことを想定して、その際に必要な `deadline:GetWidget` アクセス許可を持っていない場合に発生するものです。

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

この場合、`deadline:GetWidget` アクションを使用して *my-example-widget* リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン認証情報を提供した担当者が管理者です。

iam:PassRole を実行する権限がありません

`iam:PassRole` アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して Deadline Cloud にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

次の例のエラーは、という IAM ユーザーがコンソールを使用して Deadline Cloud でアクションを実行しようとするときに発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与されたアクセス許可が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

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

この場合、Mary のポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の 以外のユーザーに Deadline Cloud リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- Deadline Cloud がこれらの機能をサポートしているかどうかを確認するには、「」を参照してください [Deadline Cloud と IAM の連携方法](#)。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、「[IAM ユーザーガイド](#)」の「[所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する](#)」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「[サードパーティー AWS アカウント が所有する へのアクセスを提供する](#)」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイドの [外部で認証されたユーザー \(ID フェデレーション\) へのアクセスの許可](#) を参照してください。

- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、IAM ユーザーガイドの [IAM でのクロスアカウントのリソースへのアクセス](#) を参照してください。

のコンプライアンス検証 Deadline Cloud

サードパーティーの監査者は、AWS System and Organization Controls (SOC) コンプライアンスプログラム AWS Deadline Cloud の一環として のセキュリティとコンプライアンスを評価します。

AWS は、コンプライアンスプログラム AWS のサービス による対象範囲内の特定のコンプライアンスプログラムの範囲内の のリスト [AWS のサービスをコンプライアンスプログラムによる対象](#)。

サードパーティーの監査レポートは、AWS Artifact () を使用してダウンロードできますAWS Artifact。詳細については、「[AWS Artifactにレポートをダウンロードする](#)」を参照してください。

AWS コンプライアンスプログラムの詳細については、[AWS 「コンプライアンスプログラム」](#) を参照してください。

データの機密性、会社のコンプライアンス目的、適用される法律および規制によって、 を使用する際のコンプライアンス責任が決まります AWS のサービス。の使用 Deadline Cloud が SOC などの標準への準拠の対象である場合、 は以下に役立つリソース AWS を提供します。

- [セキュリティとコンプライアンスのクイックスタートガイド](#) – セキュリティとコンプライアンスに重点を置いたベースライン環境をデプロイするためのアーキテクチャ上の考慮事項と手順 AWS。
- [AWS コンプライアンスリソース](#) – 業界や地域に適用される可能性のあるワークブックとガイド。
- [AWS Config](#) – リソース設定が社内プラクティス、業界ガイドライン、規制にどの程度準拠しているかを評価します。
- [AWS Security Hub CSPM](#) – セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立つ AWS、 内のセキュリティ状態の包括的なビュー。

の耐障害性 Deadline Cloud

AWS グローバルインフラストラクチャは、AWS リージョン およびアベイラビリティゾーンを中心に構築されています。 は、低レイテンシー、高スループット、高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティゾーン AWS リージョン を提供します。アベイラビリティゾーンでは、ゾーン間で中断することなく自動的にフェールオーバーす

るアプリケーションとデータベースを設計および運用することができます。アベイラビリティゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、フォールトトレランス、および拡張性が優れています。

AWS リージョン およびアベイラビリティゾーンの詳細については、[AWS 「グローバルインフラストラクチャ」](#) を参照してください。

AWS Deadline Cloud は、ジョブアタッチメント S3 バケットに保存されているデータをバックアップしません。SAmazon S3[S3](#) バックアップメカニズムを使用して、ジョブアタッチメントデータのバックアップを有効にできます[AWS Backup](#)。

Deadline Cloud のインフラストラクチャセキュリティ

マネージドサービスである AWS Deadline Cloud は、AWS グローバルネットワークセキュリティで保護されています。AWS セキュリティサービスと [ガインフラストラクチャ AWS](#) を保護する方法については、[AWS 「クラウドセキュリティ」](#) を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して環境を AWS 設計するには、「Security Pillar AWS Well-Architected Framework」の [「Infrastructure Protection」](#) を参照してください。

AWS 公開された API コールを使用して、ネットワーク経由で Deadline Cloud にアクセスします。クライアントは次をサポートする必要があります。

- Transport Layer Security (TLS)。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードは Java 7 以降など、ほとんどの最新システムでサポートされています。

Deadline Cloud は、AWS PrivateLink Virtual Private Cloud (VPC) エンドポイントポリシーの使用をサポートしていません。エンドポイントへのフルアクセスを許可する AWS PrivateLink デフォルトのポリシーを使用します。詳細については、AWS PrivateLink ユーザーガイドの [「デフォルトのエンドポイントポリシー」](#) を参照してください。

Deadline Cloud の設定と脆弱性の分析

AWS は、ゲストオペレーティングシステム (OS) やデータベースのパッチ適用、ファイアウォール設定、ディザスタリカバリなどの基本的なセキュリティタスクを処理します。これらの手順は適切な第三者によって確認され、証明されています。詳細については、以下のリソースを参照してください。

- [責任共有モデル](#)
- [Amazon Web Services: セキュリティプロセスの概要](#) (ホワイトペーパー)

AWS Deadline Cloud は、サービスマネージドフリートまたはカスターマネージドフリートのタスクを管理します。

- サービスマネージドフリートの場合、Deadline Cloud はゲストオペレーティングシステムを管理します。
- カスターマネージドフリートの場合、オペレーティングシステムを管理する責任があります。

AWS Deadline Cloud の設定と脆弱性の分析の詳細については、「」を参照してください。

- [Deadline Cloud のセキュリティのベストプラクティス](#)

サービス間での不分別な代理処理の防止

混乱した代理問題は、アクションを実行するためのアクセス許可を持たないエンティティが、より特権のあるエンティティにアクションの実行を強制できてしまう場合に生じる、セキュリティ上の問題です。では AWS、サービス間のなりすましにより、混乱した代理問題が発生する可能性があります。サービス間でのなりすましは、1つのサービス (呼び出し元サービス) が、別のサービス (呼び出し対象サービス) を呼び出すときに発生する可能性があります。呼び出し元サービスは、本来ならアクセスすることが許可されるべきではない方法でその許可を使用して、別のお客様のリソースに対する処理を実行するように操作される場合があります。これを防ぐため、AWS では、アカウントのリソースへのアクセス権が付与されたサービスプリンシパルで、すべてのサービスのデータを保護するために役立つツールを提供しています。

リソースポリシーで [aws:SourceArn](#) および [aws:SourceAccount](#) グローバル条件コンテキストキーを使用して、ガリソースに別のサービス AWS Deadline Cloud に付与するアクセス許可を制限することをお勧めします。クロスサービスアクセスにリソースを1つだけ関連付けたい場合は、aws:SourceArn を使用します。そのアカウント内のリソースをクロスサービスの使用に関連付けることを許可する場合は、aws:SourceAccount を使用します。

「混乱した代理」問題から保護するための最も効果的な方法は、リソースの完全な Amazon リソースネーム (ARN) を指定しながら、グローバル条件コンテキストキー aws:SourceArn を使用することです。リソースの完全な ARN が不明な場合や、複数のリソースを指定する場合には、グローバルコンテキスト条件キー aws:SourceArn で、ARN の未知部分を示すためにワイルドカード文字 (*) を使用します。例えば、arn:aws:deadline:*:**123456789012**:*。

aws:SourceArn の値に Amazon S3 バケット ARN などのアカウント ID が含まれていない場合は、両方のグローバル条件コンテキストキーを使用して、アクセス許可を制限する必要があります。

次の例は、で aws:SourceArn および aws:SourceAccount グローバル条件コンテキストキーを使用して、混乱した代理問題 Deadline Cloud を防ぐ方法を示しています。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "deadline.amazonaws.com"
    },
    "Action": "deadline:CreateFarm",
    "Resource": [
      "*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:deadline:*:111122223333:"
      },
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      }
    }
  }
}
```

インターフェイスエンドポイント (AWS PrivateLink) AWS Deadline Cloud を使用した へのアクセス

を使用して AWS PrivateLink、VPC と の間にプライベート接続を作成できます AWS Deadline Cloud。インターネットゲートウェイ、NAT デバイス、VPN 接続、または Direct Connect 接続を使用せずに、VPC 内にある Deadline Cloud のように にアクセスできます。VPC 内のインスタンスは Deadline Cloud にアクセスするためにパブリック IP アドレスを必要としません。

このプライベート接続を確立するには、AWS PrivateLinkを利用したインターフェイスエンドポイントを作成します。インターフェイスエンドポイントに対して有効にする各サブネットにエンドポイントネットワークインターフェイスを作成します。これらは、Deadline Cloud宛でのトラフィックのエントリーポイントとして機能するリクエスト管理型ネットワークインターフェイスです。

Deadline Cloud には、デュアルスタックのエンドポイントも用意されています。デュアルスタックエンドポイントは、IPv6 および IPv4 経由のリクエストをサポートします。

詳細については「AWS PrivateLink ガイド」の「[Access AWS のサービス through AWS PrivateLink](#)」を参照してください。

に関する考慮事項 Deadline Cloud

のインターフェイスエンドポイントを設定する前に Deadline Cloud、「AWS PrivateLink ガイド」の「[インターフェイス VPC エンドポイントを使用して AWS サービスにアクセスする](#)」を参照してください。

Deadline Cloud は、インターフェイスエンドポイントを介したすべての API アクションの呼び出しをサポートしています。

デフォルトでは、へのフルアクセス Deadline Cloud はインターフェイスエンドポイントを介して許可されます。または、セキュリティグループをエンドポイントネットワークインターフェイスに関連付けて、インターフェイスエンドポイント Deadline Cloud 経由でのトラフィックを制御することもできます。

Deadline Cloud は VPC エンドポイントポリシーもサポートしています。詳細については、『AWS PrivateLink ガイド』の「[Control access to VPC endpoints using endpoint policies \(エンドポイントポリシーを使用して VPC エンドポイントへのアクセスをコントロールする\)](#)」を参照してください。

Deadline Cloud エンドポイント

Deadline Cloud は、IPv4 には 4 AWS PrivateLink つのエンドポイントを使用し、IPv4IPv6 には 2 つのエンドポイントを使用します。

ワーカーは `scheduling.deadline.region.amazonaws.com` エンドポイントを使用して、キューからタスクを取得し、進捗状況を報告し Deadline Cloud、タスク出力を送り返します。カスタマーマネージドフリートを使用している場合、管理オペレーションを使用しない限り、作成する必要のあるのはスケジューリングエンドポイントだけです。たとえば、ジョブがより多くのジョブを作成する場合は、管理エンドポイントを有効にして `CreateJob` オペレーションを呼び出す必要があります。

Deadline Cloud モニターは を使用して、キューとフリートの作成と変更、ジョブ、ステップ、タスクのリストの取得など、ファーム内のリソースmanagement.deadline.*region*.amazonaws.comを管理します。

AWS SDKsと CLI は、 managementおよび schedulingプレフィックスをエンドポイントに自動的に追加します。この動作を無効にする場合は、「 SDK およびツールリファレンスガイド」の「[ホストプレフィックスインジェクション](#)」セクションを参照してください。AWS SDKs

Deadline Cloud には、次の AWS サービスエンドポイントのエンドポイントも必要です。

- インターネット接続のないサブネットにカスタマーマネージドフリートをセットアップする場合は、ワーカーがログを書き込めるように、Amazon CloudWatch Logs の VPC エンドポイントを作成する必要があります。詳細については、「[Amazon CloudWatch によるモニタリング](#)」を参照してください。
- ジョブアタッチメントを使用する場合は、ワーカーがアタッチメントにアクセスできるように、Amazon Simple Storage Service (Amazon S3) の VPC エンドポイントを作成する必要があります。詳細については、「[のジョブアタッチメント Deadline Cloud](#)」を参照してください。

のエンドポイントを作成する Deadline Cloud

Amazon VPC コンソールまたは AWS Command Line Interface () Deadline Cloud を使用して、 のインターフェイスエンドポイントを作成できますAWS CLI。詳細については、「AWS PrivateLink ガイド」の「[インターフェイスエンドポイントを作成](#)」を参照してください。

次のサービス名 Deadline Cloud を使用して、 の管理エンドポイントとスケジューリングエンドポイントを作成します。*region* をデプロイした AWS リージョン に置き換えます Deadline Cloud。

```
com.amazonaws.region.deadline.management
```

```
com.amazonaws.region.deadline.scheduling
```

Deadline Cloud はデュアルスタックのエンドポイントをサポートしています。

インターフェイスエンドポイントのプライベート DNS を有効にすると、デフォルトのリージョン DNS 名 Deadline Cloud を使用して に API リクエストを行うことができます。たとえば、ワーカーオペレーションscheduling.deadline.us-east-1.amazonaws.comの場合は、その他すべてのオペレーションmanagement.deadline.us-east-1.amazonaws.comの場合は です。

カスタマーマネージドフリートがインターネット接続のないサブネット上にある場合は、次のサービス名を使用して CloudWatch Logs エンドポイントを作成する必要があります。

```
com.amazonaws.region.logs
```

ジョブアタッチメントを使用してファイルを転送する場合は、次のサービス名を使用して Amazon S3 エンドポイントを作成する必要があります。

```
com.amazonaws.region.s3
```

制限されたネットワーク環境

Deadline Cloud には、アーティストや他のユーザーがローカルワークステーションで使用するツールが用意されています。これらのツールでは、関数を実行するために AWS API エンドポイントとウェブエンドポイントにアクセスする必要があります。次世代ファイアウォール (NGFW) や Secure Web Gateway (SWG) などのウェブコンテンツフィルタリングソリューションを使用して特定の AWS ドメインまたは URL エンドポイントへのアクセスをフィルタリングする場合は、ウェブコンテンツフィルタリングソリューションの許可リストに次のドメインまたは URL エンドポイントを追加する必要があります。

AWS 許可リストの API エンドポイント

、モニター AWS マネジメントコンソール、CLI、統合送信者などの Deadline Cloud クライアントツールには、Deadline Cloud に加えて AWS APIs へのアクセスが必要です。これらのエンドポイントは IPv4 のみをサポートします。

- `scheduling.deadline.[Region].amazonaws.com`
- `management.deadline.[Region].amazonaws.com`
- `logs.[Region].amazonaws.com`
- `ec2.[Region].amazonaws.com`
- `s3.[Region].amazonaws.com`
- `sts.[Region].amazonaws.com`
- `identitystore.[Region].amazonaws.com`

許可リストのウェブドメイン

Deadline Cloud モニターを操作するには、次のドメインにアクセスする必要があります。

AWS サインインのドメインの許可リストの詳細については、AWS 「サインインユーザーガイド」の「[許可リストに追加するドメイン](#)」を参照してください。

- `downloads.deadlinecloud.amazonaws.com`
- `d2ev1rdnjzhmnr.cloudfront.net`
- `prod.log.shortbread.aws.dev`
- `prod.tools.shortbread.aws.dev`
- `prod.log.shortbread.analytics.console.aws.a2z.com`
- `prod.tools.shortbread.analytics.console.aws.a2z.com`
- `global.help-panel.docs.aws.a2z.com`
- `[Region].signin.aws`
- `[Region].signin.aws.amazon.com`
- `sso.[Region].amazonaws.com`
- `portal.sso.[Region].amazonaws.com`
- `oidc.[Region].amazonaws.com`
- `assets.sso-portal.[Region].amazonaws.com`

許可リストを作成する環境固有のエンドポイント

これらのドメインは、Deadline Cloud の特定の設定によって異なります。追加の Deadline Cloud モニターまたはキューが作成された場合は、追加のドメインを許可リストに登録する必要があります。

- `[Directory ID or alias].awsapps.com`

このドメインは IAM Identity Center のセットアップに関連付けられており、同じ IAM Identity Center インスタンスを使用するこのすべてのセットアップで同じである必要があります。正確な値は、IAM Identity Center コンソールの設定 → AWS アクセスポータル URL のエンタープライズ管理者が見つけることができます。

- `[Monitor alias].[Region].deadlinecloud.amazonaws.com`

このドメインは Deadline Cloud の Monitor セットアップ用です。アーティストは、ブラウザまたは Deadline Cloud Monitor アプリケーションにこのリンクを入力します。Deadline Cloud が将来的に追加のアカウントまたはリージョンに設定されている場合、このドメインは変更されます。この値は、Dashboard → Monitor overview → Monitor details → URL の Deadline Cloud コンソールにあります。

- `[Bucket name].[Region].s3.amazonaws.com`

これは、Deadline Cloud キューで使用されるジョブアタッチメントバケットのドメインです。各キューには、独自のジョブアタッチメントバケットを設定できます。正確なバケット名は、Deadline Cloud コンソールの Queues → Queue details → Job attachments にあります。ジョブアタッチメントの詳細については、キューのドキュメントを参照してください。

Deadline Cloud のセキュリティのベストプラクティス

AWS Deadline Cloud (Deadline Cloud) には、独自のセキュリティポリシーを開発および実装する際に考慮すべき多くのセキュリティ機能が用意されています。以下のベストプラクティスは一般的なガイドラインであり、完全なセキュリティソリューションを説明するものではありません。これらのベストプラクティスはお客様の環境に適切ではないか、十分ではない場合があるため、これらは指示ではなく、有用な考慮事項と見なしてください。

Note

多くのセキュリティトピックの重要性の詳細については、[「責任共有モデル」](#)を参照してください。

データ保護

データ保護の目的で、(AWS Identity and Access Management IAM) を使用して AWS アカウント 認証情報を保護し、個々のアカウントを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。

- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。
- AWS 暗号化ソリューションと、内のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon S3 (Amazon Simple Storage Service) に保存されている個人情報の発見と保護を支援する Amazon Macie などのアドバンスドマネージドセキュリティサービスを使用します。
- コマンドラインインターフェイスまたは API を使用して AWS にアクセスするときに FIPS 140-2 検証済みの暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-2](#)」を参照してください。

顧客のアカウント番号などの機密の識別情報は、[Name] (名前) フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。この推奨事項には、コンソール、API、AWS CLI または AWS SDKs AWS のサービス を使用して AWS Deadline Cloud または他の を使用する場合があります。Deadline Cloud または他の サービスに入力したデータは、診断ログに取り込まれる可能性があります。外部サーバーへの URL を指定するときは、そのサーバーへのリクエストを検証するための認証情報を URL に含めないでください。

AWS Identity and Access Management アクセス許可

ユーザー、AWS Identity and Access Management (IAM) ロール、およびユーザーに最小権限を付与して、AWS リソースへのアクセスを管理します。AWS アクセス認証情報を作成、配布、ローテーション、および取り消すための認証情報管理ポリシーと手順を確立します。詳細については、「IAM ユーザーガイド」の「[IAM のベストプラクティス](#)」を参照してください。

ユーザーおよびグループとしてジョブを実行する

Deadline Cloud でキュー機能を使用する場合、OS ユーザーがキューのジョブに対する最小特権のアクセス許可を持つように、オペレーティングシステム (OS) ユーザーとそのプライマリグループを指定するのがベストプラクティスです。

「ユーザーとして実行」(およびグループ) を指定すると、キューに送信されたジョブのプロセスは、その OS ユーザーを使用して実行され、そのユーザーの関連する OS アクセス許可を継承します。

フリートとキューの設定を組み合わせ、セキュリティ体制を確立します。キュー側では、「ユーザーとして実行されるジョブ」と IAM ロールを指定して、キューのジョブに OS と AWS アクセス許可を使用できます。フリートは、特定のキューに関連付けられているときにキュー内でジョブを

実行するインフラストラクチャ (ワーカーホスト、ネットワーク、マウントされた共有ストレージ) を定義します。ワーカーホストで使用可能なデータは、1 つ以上の関連付けられたキューのジョブによってアクセスされる必要があります。ユーザーまたはグループを指定すると、ジョブ内のデータを他のキュー、インストールされている他のソフトウェア、またはワーカーホストにアクセスできる他のユーザーから保護できます。キューにユーザーがない場合、キューはエージェントユーザーとして実行され、任意のキューユーザーを偽装 (sudo) できます。このようにして、ユーザーのないキューは権限を別のキューにエスカレートできます。

ネットワーク

トラフィックが傍受またはリダイレクトされないようにするには、ネットワークトラフィックをルーティングする方法と場所を保護することが重要です。

ネットワーク環境は、次の方法で保護することをお勧めします。

- Amazon Virtual Private Cloud (Amazon VPC) サブネットルートテーブルを保護して、IP レイヤートラフィックのルーティング方法を制御します。
- ファームまたはワークステーションのセットアップで Amazon Route 53 (Route 53) を DNS プロバイダーとして使用している場合は、Route 53 API への安全なアクセスを確保します。
- オンプレミスのワークステーションやその他のデータセンターを使用して AWS の外部で Deadline Cloud に接続する場合は、オンプレミスのネットワークインフラストラクチャを保護します。これには、ルーター、スイッチ、その他のネットワークデバイスの DNS サーバーとルートテーブルが含まれます。

ジョブとジョブデータ

Deadline Cloud ジョブは、ワーカーホストのセッション内で実行されます。各セッションはワーカーホストで 1 つ以上のプロセスを実行します。通常、出力を生成するにはデータを入力する必要があります。

このデータを保護するには、キューを使用してオペレーティングシステムユーザーを設定できます。ワーカーエージェントは、キュー OS ユーザーを使用してセッションサブプロセスを実行します。これらのサブプロセスは、キュー OS ユーザーのアクセス許可を継承します。

これらのサブプロセスアクセスのデータへのアクセスを保護するために、ベストプラクティスに従うことをお勧めします。詳細については、「[責任共有モデル](#)」を参照してください。

ファーム構造

Deadline Cloud フリートとキューは、さまざまな方法で配置できます。ただし、特定の手配にはセキュリティ上の影響があります。

ファームには最も安全な境界の 1 つがあります。これは、Deadline Cloud リソースをフリート、キュー、ストレージプロファイルなどの他のファームと共有できないためです。ただし、ファーム内で外部 AWS リソースを共有できるため、セキュリティの境界が侵害されます。

適切な設定を使用して、同じファーム内のキュー間にセキュリティ境界を確立することもできます。

同じファームに安全なキューを作成するには、次のベストプラクティスに従います。

- フリートを同じセキュリティ境界内のキューにのみ関連付けます。次の点に注意してください。
 - ワーカーホストでジョブが実行された後、一時ディレクトリやキューユーザーのホームディレクトリなどにデータが残される可能性があります。
 - ジョブの送信先のキューに関係なく、同じ OS ユーザーがサービス所有のフリートワーカーホストですべてのジョブを実行します。
 - ジョブがワーカーホストで実行されているプロセスから離れ、他のキューのジョブが実行中の他のプロセスを監視できる場合があります。
- 同じセキュリティ境界内のキューのみがジョブアタッチメントの Amazon S3 バケットを共有していることを確認します。
- 同じセキュリティ境界内のキューのみが OS ユーザーを共有していることを確認します。
- ファームに統合されている他の AWS リソースを境界に保護します。

ジョブアタッチメントキュー

ジョブアタッチメントは、Amazon S3 バケットを使用するキューに関連付けられています。

- ジョブアタッチメントは、Amazon S3 バケットのルートプレフィックスとの間で書き込みおよび読み取りを行います。CreateQueue API コールでこのルートプレフィックスを指定します。
- バケットには対応する `Queue Role`、キューユーザーにバケットとルートプレフィックスへのアクセスを許可するロールを指定します。キューを作成するときは、ジョブアタッチメントバケットとルートプレフィックスとともに `Queue Role Amazon` リソースネーム (ARN) を指定します。

- AssumeQueueRoleForRead、および AssumeQueueRoleForWorker API オペレーションへの認可された呼び出しは AssumeQueueRoleForUser、の一時的なセキュリティ認証情報のセットを返します Queue Role。

キューを作成し、Amazon S3 バケットとルートプレフィックスを再利用すると、情報が権限のない当事者に開示されるリスクがあります。たとえば、QueueA と QueueB は同じバケットとルートプレフィックスを共有します。安全なワークフローでは、ArtistA は QueueA にアクセスできますが、QueueB にはアクセスできません。ただし、複数のキューがバケットを共有する場合、ArtistA は QueueB データ内のデータにアクセスできます。 QueueA

コンソールは、デフォルトで安全なキューを設定します。キューが共通のセキュリティ境界の一部でない限り、Amazon S3 バケットとルートプレフィックスの個別の組み合わせがあることを確認します。

キューを分離するには、バケットとルートプレフィックスへのキューアクセスのみを許可する Queue Role ように を設定する必要があります。次の例では、各 **#####** をリソース固有の情報に置き換えます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME",
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME/JOB_ATTACHMENTS_ROOT_PREFIX/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        }
      }
    }
  ]
}
```

```
    }
  },
  {
    "Action": [
      "logs:GetLogEvents"
    ],
    "Effect": "Allow",
    "Resource": "arn:aws:logs:us-east-1:111122223333:log-group:/aws/
deadline/FARM_ID/*"
  }
]
```

また、ロールに信頼ポリシーを設定する必要があります。次の例では、#####テキストをリソース固有の情報に置き換えます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "sts:AssumeRole"
      ],
      "Effect": "Allow",
      "Principal": {
        "Service": "deadline.amazonaws.com"
      },
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnEquals": {
          "aws:SourceArn": "arn:aws:deadline:us-
east-1:111122223333:farm/FARM_ID"
        }
      }
    },
    {
      "Action": [
```

```
        "sts:AssumeRole"
    ],
    "Effect": "Allow",
    "Principal": {
        "Service": "credentials.deadline.amazonaws.com"
    },
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "111122223333"
        },
        "ArnEquals": {
            "aws:SourceArn": "arn:aws:deadline:us-east-1:111122223333:farm/FARM_ID"
        }
    }
}
]
```

カスタムソフトウェア Amazon S3 バケット

に次のステートメントを追加してQueue Role、Amazon S3 バケット内のカスタムソフトウェアにアクセスできます。次の例では、**Software_BUCKET_NAME** を S3 バケットの名前に置き換え、**BUCKET_ACCOUNT_OWNER** をバケットを所有する AWS アカウント ID に置き換えます。

```
"Statement": [
  {
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME",
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME/*"
    ],
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "BUCKET_ACCOUNT_OWNER"
      }
    }
  }
]
```

]

Amazon S3 セキュリティのベストプラクティスの詳細については、Amazon Simple Storage Service ユーザーガイドの「Amazon [Amazon S3 のセキュリティのベストプラクティス](#)」を参照してください。

ワーカーホスト

ワーカーホストを保護して、各ユーザーが割り当てられたロールに対してのみオペレーションを実行できるようにします。

ワーカーホストを保護するには、次のベストプラクティスをお勧めします。

- ホスト設定スクリプトを使用すると、ワーカーのセキュリティとオペレーションが変更される可能性があります。設定が正しくないと、ワーカーが不安定になったり、動作が停止したりする可能性があります。このような障害をデバッグするのはお客様の責任です。
- これらのキューに送信されたジョブが同じセキュリティ境界内にある場合を除き、複数のキューで同じ `jobRunAsUser` 値を使用しないでください。
- ワーカーエージェントが実行する OS ユーザーの名前 `jobRunAsUser` にキューを設定しないでください。
- 目的のキューワークロードに必要な最小特権の OS アクセス許可をキューユーザーに付与します。エージェントプログラムファイルやその他の共有ソフトウェアを操作するためのファイルシステムの書き込みアクセス許可がないことを確認します。
- のルートユーザー `Linux` と `Administrator` が所有するアカウントのみが `Windows` 独自のものであり、ワーカーエージェントプログラムファイルを変更できることを確認します。
- `Linux` ワーカーホストでは、ワーカーエージェントユーザーがキューユーザーとしてプロセスを起動 `/etc/sudoers` できるようにする `umask` オーバーライドを で設定することを検討してください。この設定は、他のユーザーがキューに書き込まれたファイルにアクセスできないようにするのに役立ちます。
- 信頼できる個人にワーカーホストへの最小特権アクセスを付与します。
- ローカル DNS オーバーライド設定ファイル (`/etc/hosts Linux` および `C:\Windows\system32\etc\hosts`) へのアクセス許可を制限 `Windows` し、ワークステーションとワーカーホストオペレーティングシステムにテーブルをルーティングします。
- ワークステーションとワーカーホストオペレーティングシステムの DNS 設定へのアクセス許可を制限します。

- オペレーティングシステムとインストールされているすべてのソフトウェアに定期的にパッチを適用します。このアプローチには、送信者、アダプター、ワーカーエージェント、OpenJDパッケージなど、Deadline Cloud で特に使用されるソフトウェアが含まれます。
- Windows キュー に強力なパスワードを使用します `jobRunAsUser`。
- キュー のパスワードを定期的にローテーションします `jobRunAsUser`。
- Windows パスワードシークレットへの最小特権アクセスを確保し、未使用のシークレットを削除します。
- 今後実行する `jobRunAsUser` スケジュールコマンドをキューに付与しないでください。
 - で Linux、これらのアカウントによる `cron` および `at` へのアクセスを拒否します。
 - で Windows、これらのアカウントによる Windows タスクスケジューラへのアクセスを拒否します。

Note

オペレーティングシステムとインストールされたソフトウェアに定期的にパッチを適用する重要性の詳細については、[「責任共有モデル」](#)を参照してください。

ホスト設定スクリプト

- ホスト設定スクリプトを使用すると、ワーカーのセキュリティとオペレーションが変更される可能性があります。設定が正しくないと、ワーカーが不安定になったり、動作が停止したりする可能性があります。このような障害をデバッグするのはお客様の責任です。

ワークステーション

Deadline Cloud にアクセスできるワークステーションを保護することが重要です。このアプローチは、Deadline Cloud に送信するジョブが、に請求される任意のワークロードを実行できないようにするのに役立ちます AWS アカウント。

アーティストワークステーションを保護するには、次のベストプラクティスをお勧めします。詳細については、[責任共有モデル](#)を参照してください。

- Deadline Cloud など AWS、へのアクセスを提供する永続的な認証情報を保護します。詳細については、「IAM ユーザーガイド」の [IAM ユーザーのアクセスキーの管理](#) を参照してください。

- 信頼できる安全なソフトウェアのみをインストールします。
- ユーザーは ID プロバイダーとフェデレーションし、一時的な認証情報 AWS を使用して にアクセスする必要があります。
- Deadline Cloud 送信者プログラムファイルに対する安全なアクセス許可を使用して、改ざんを防止します。
- 信頼できる個人にアーティストワークステーションへの最小特権アクセスを付与します。
- Deadline Cloud Monitor を通じて取得した送信者とアダプターのみを使用してください。
- ローカル DNS オーバーライド設定ファイル (/etc/hosts Linux および、C:\Windows\system32\etc\hosts) へのアクセス許可を制限 Windows し macOS、ワークステーションとワーカーホストオペレーティングシステムにテーブルをルーティングします。
- ワークステーションとワーカーホストオペレーティングシステム/etc/resolve.confのアクセス許可を に制限します。
- オペレーティングシステムとインストールされているすべてのソフトウェアに定期的にパッチを適用します。このアプローチには、送信者、アダプター、ワーカーエージェント、OpenJD パッケージなど、Deadline Cloud で特に使用されるソフトウェアが含まれます。

ダウンロードしたソフトウェアの信頼性を検証する

インストーラをダウンロードした後、ファイルの改ざんから保護するために、ソフトウェアの信頼性を確認します。この手順は、Windows および Linux システムの両方で機能します。

Windows

ダウンロードしたファイルの真正性を検証するには、次の手順を実行します。

1. 次のコマンドで、 を、検証するファイル *file* に置き換えます。例えば、 **C:\PATH\TO\MY\DeadlineCloudSubmitter-windows-x64-installer.exe** 。また、 を、インストールされている SignTool SDK のバージョン *signtool-sdk-version* に置き換えます。例えば、 **10.0.22000.0** 。

```
"C:\Program Files (x86)\Windows Kits\10\bin\signtool-sdk-version\x86\signtool.exe" verify /vfile
```

2. たとえば、次のコマンドを実行して Deadline Cloud 送信者インストーラファイルを確認できます。

```
"C:\Program Files (x86)\Windows Kits\10\bin
\10.0.22000.0\x86\signtool.exe" verify /v DeadlineCloudSubmitter-
windows-x64-installer.exe
```

Linux

ダウンロードしたファイルの真正性を確認するには、`gpg` コマンドラインツールを使用します。

1. 次のコマンドを実行してOpenPGPキーをインポートします。

```
gpg --import --armor <<EOF
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBG1ANDUBEACg6zffjN43gqe5ryPhk+wQM10rEdvmItw4WPWaVsN+/at/OIJw
MGCagSYXcgR+jKbsHQ0QoEQdo5SrxHjpkTEs3KQhGvf+ehrU1Ac7koXKIBWtes+
BI9F0s1RECz0nXT0y/cd/90RXjpf07mreTLIKNIbybULfad82nYykpITjFr5XRGj
/shYkucxRQZdwkgkIYyV25pPICPd2RsX+Zua85jV8mCqVffDfRXvgcPe3+ofClj/
2CE8UfUIq08Csu4YEKsqR3aeoT0EFT4kuQR5nFXVzor0EkQt03gB35KNWKM1IOU
2vA+wyoL7nWSii4yfYtW3EZ+3gq6HxvnT9Zs8MC53uT0i0damASXecYREwGmY/io
6n5XTEA/35LNB14A756vSTZ7h4VFJAN5BpuqxstI1D7ou94skoSmcPoC/iniTvY9
kZy1U50CH/nifMAHM2a5jrQe180cW4oko9eyc8ENQpSy15JE1F0KFF7D/4tcZJLF
F0VBTXbhfvq3dPfoq94Iwt7p540vwj0S//CEu3jZYbN12QC/3YiHE2H2XyGCQbq6
2MjcuxLnEapoRIqfbi8GPtCWVPzm28WgyKIDofWICczzeJFFJnvzrY3wRG64ibKJ
bR/uedwua1UuiC482V1FD5ffmzSSs8ktTp9hgj7RGDX1c9NTcF1jHxG9hwARAQAB
tCxBV1MgRGVhZGxpbnUgQ2xvdWQgPGF3cy1kZWFKbGluZUBhbWF6b24uY29tPokC
VwQTAQgAQRyhbJmXd7So2csyehiIYsg71N18bhtjBQJpQDQ1AhsvBQkDwmcABQsJ
CAcCAiICBhUKCQgLAQwAgMBAh4HAheAAAoJEMg71N18bhtjk2UP/3h4K1EzZ0/7
BxRmkbixuo1Quq0GvA6tXbSWaM8QH5jglcvL12PZLALk1LT4v82uCsLR11F8/Tch
cC10SZE0FIS+XxAaw1Xfai6jlyLhab0wKF2ylq5eJ1Lcw11h2nAArDRb4fLD0m1g
Dfquetq/XEpyXp0SkWxGRV4R1UdjQfytxrncUnsT5/fk5f9VdDbLu6K/1EmwfyYjB
1Xv0uUCkqPot0Smbv0h3PY3Hi3n54ncy8NfTeV+TUvSe3C1s1zN18aqHoTxJB/eU
kp+LFZ9m+igpSYnKeg1Knyty1H3KGCjThglT/QXnI1wNTqmj1kFBVwtt/y1mtnA+
CPIUHP1CtbKsHaltp411Bm5TVtPN/Wqqicn5QL14khg7R4K+V2aaA4ubY6p1tG9
0fFhN5tTnHDSKWMfmb83wfh5Zkcg85c3egjoit+wgQRAQVqbznx7NqAHs9VoDIu
SPcAr+C329A0Bzod4gyNGH7Ah5DkMITo404+axnAU9yhFOHcMjMTIask/fNg1Aum
OqYPMUwcv1GZjLaTJyfGGC1xALsYR0KHnwIehD06MHR/Z98bGkcV8+Y0q8UPsd1
VN1fc1rjCJh/AT3w6owvG4DaEwspseSjzHv16mW4e2N6Uu23SPzqQsJ5qYN2g8D+
P7N9LGDfP8DaYc5JM9mlyFmYI2Q94ufl
=rY5l
-----END PGP PUBLIC KEY BLOCK-----
EOF
```

2. OpenPGP キーを信頼するかどうかを決定します。上記のキーを信頼するかどうかを決定する際に考慮すべき要素には、次のようなものがあります。
 - このウェブサイトから GPG キーを取得するために使用したインターネット接続は安全です。
 - このウェブサイトアクセスするデバイスは安全です。
 - AWS は、このウェブサイトでの OpenPGP パブリックキーのホスティングを保護するための対策を講じました。
3. OpenPGP キーを信頼する場合は、次の例 gpg のように で信頼するようにキーを編集します。

```
$ gpg --edit-key 0xB840C08C29A90796A071FAA5F6CD3CE6B76F3CEF

gpg (GnuPG) 2.0.22; Copyright (C) 2013 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

pub 4096R/4BF0B8D2  created: 2023-06-23  expires: 2025-06-22  usage: SCEA
                        trust: unknown      validity: unknown
[ unknown] (1). AWS Deadline Cloud example@example.com

gpg> trust
pub 4096R/4BF0B8D2  created: 2023-06-23  expires: 2025-06-22  usage: SCEA
                        trust: unknown      validity: unknown
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com

Please decide how far you trust this user to correctly verify other users'
keys
(by looking at passports, checking fingerprints from different sources,
etc.)

 1 = I don't know or won't say
 2 = I do NOT trust
 3 = I trust marginally
 4 = I trust fully
 5 = I trust ultimately
m = back to the main menu

Your decision? 5
Do you really want to set this key to ultimate trust? (y/N) y
```

```
pub 4096R/4BF0B8D2 created: 2023-06-23 expires: 2025-06-22 usage: SCEA
trust: ultimate validity: unknown
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com
Please note that the shown key validity is not necessarily correct
unless you restart the program.

gpg> quit
```

4. Deadline Cloud 送信者インストーラを検証する

Deadline Cloud 送信者インストーラを確認するには、次の手順を実行します。

- a. Deadline Cloud 送信者インストーラの署名ファイルをダウンロードします。

[署名ファイル \(.sig\) をダウンロードする](#)

- b. 以下を実行して、Deadline Cloud 送信者インストーラの署名を確認します。

```
gpg --verify ./DeadlineCloudSubmitter-linux-x64-installer.run.sig ./
DeadlineCloudSubmitter-linux-x64-installer.run
```

5. Deadline Cloud モニターを確認する

Note

署名ファイルまたはプラットフォーム固有の方法を使用して、Deadline Cloud モニターのダウンロードを確認できます。プラットフォーム固有の方法については、Linux (Debian) タブ、Linux (RPM) タブ、またはダウンロードしたファイルタイプに基づく Linux (ApplImage) タブを参照してください。

署名ファイルを使用して Deadline Cloud Monitor デスクトップアプリケーションを検証するには、次の手順を実行します。

- a. Deadline Cloud Monitor インストーラに対応する署名ファイルをダウンロードします。

- [.deb 署名ファイルをダウンロードする](#)
- [.rpm 署名ファイルをダウンロードする](#)
- [.ApplImage 署名ファイルをダウンロードする](#)

- b. 署名を確認します。

.deb の場合:

```
gpg --verify ./deadline-cloud-monitor_amd64.deb.sig ./deadline-cloud-monitor_amd64.deb
```

.rpm の場合:

```
gpg --verify ./deadline-cloud-monitor.x86_64.rpm.sig ./deadline-cloud-monitor.x86_64.rpm
```

.AppImage の場合:

```
gpg --verify ./deadline-cloud-monitor_amd64.AppImage.sig ./deadline-cloud-monitor_amd64.AppImage
```

- c. 出力が次のようになっていることを確認します。

```
gpg: Signature made Mon Apr 1 21:10:14 2024 UTC
```

```
gpg: using RSA key B840C08C29A90796A071FAA5F6CD3CE6B7
```

出力に というフレーズが含まれている場合 Good signature from "AWS Deadline Cloud"、署名が正常に検証され、Deadline Cloud Monitor のインストールスクリプトを実行できます。

履歴キー

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
mQINBGX6GQsBEADduUtJgqSXI+q7606fsFwEYKmbnlyL0xKvlq32EZuyv0otZo5L  
le4m5Gg52AzrvPvDiUTLooAlvYeozaYyirIGsK08Ydz0Ftdjroiuh/mw9JSJDJRI  
rnRn5yKet1JFezkjopA3pjsTBP6lW/mb1bDBDEwwwtH0x9lV7A03FJ9T7Uzu/qSh  
q0/UYdkafro3cPASvkqgDt2tCvURfBcUCAjZVFcLZcVD5iwXacxvKsxxS/e7kuVV  
I1+VGT8Hj8XzWYhjCZx0LZk/fvpYPMYEEUjN0fYUp6RtMIXve0C9awwMCy5nBG2J  
eE2015DsCpTaBd4Fdr3LWcSs8JFA/YfP9auL3Ncz0ozPoVJt+fw8CB1VIX00J715  
hvHDjcC+5v0wxqAlMG6+f/SX7CT8FXK+L3i0J5gBYUNXqHSxUdv8kt76/KVmQa1B  
Ak1+MPKpMq+1hw++S3G/lXqwWaDNQbRRw7dSZHymQVXvPp1nscq3hV7K10M+6s6g  
1g4mvFY4l1f6DhptwZLWYQXU8rBQpojvQfiSmDFrFPWF5BexesuVnkGIo1Qok1Kx  
AVUSdJPVEJCTeyy7td4FPhBaSqT5vW3+ANbr9b/uoRYWJvn17dN0cc9HuRh/Ai+I
```

```
nkfECo2WUDLZ0fEKGjGyFX+todWvJXjvc5kmE9Ty5vJp+M9Vvb8jd6t+mwARAQAB
tCxBV1MgRGVhZGxpbnUgQ2xvdWQgPGF3cy1kZWFKbGluZUBhbWF6b24uY29tPokC
VwQTAQgAQRyYhBLhAwIwpqQeWoHH6pfbNP0a3bzzvBQJ1+hkLAXsvBAUJA8JnAAUL
CQgHAgIiAgYVCgkICwIDFgIBAh4HAheAAAoJEPbNP0a3bzzvKswQAJXzKSAY8sY8
F6Eas2oYwIDDdDurs8FiEnFghjUE06MTt9AykF/jw+CQg2UzFtEy0bHBymghmhXE
3buVeom96tgM3ZDfZu+sxi5pGX6oAQnZ6riztN+VpkpQmLgwtMGpSML13KLwnv2k
WK8mrR/fPMkfaewB7A6RIUYiW33GAL4KfMIIs8/vIwIJw99NxHpZQVoU6dFpuDtE
10uxGcCqGJ7mAmo6H/YawSNp2Ns80gyqIKYo7o3LJ+WRroIRlQyctq8gnR9JvYXX
42ASqLq5+0XKo4qh81b1XKYqtc176BbbSNFjWnzIQgKDgNiHFZCdc0VgqDhw015r
NICbqqwNLj/Fr2kecYx180Ktp10j00w5I0yh3bf3MVGWnYRdjvA1v+/CO+55N4g
z0kf50Lcdu5RtqV10XBCifn28pecqPaSdYcssYSR15DLiFktGbNzTGcZZwITTKQc
af8PPdTGtnnb6P+cdbW3bt9MvtN5/dgSHLThnS8MPEuNCtkTnpXshuVuBGgwBMdb
qUC+HjqvhZzbwns8dr5WI+6HWNBFgGANn6ageY158vVp0UkuNP8wcWjRARciHXZx
ku6W2jPTHDWGNrBQ02Fx7fd2QYJheIPPAShHcfJ0+XgWcof45D0vAxAJ8gGg9Eq+
gFWhsx4NSHn2gh1gDZ410u/4exJ11wPM
=uVaX
-----END PGP PUBLIC KEY BLOCK-----
EOF
```

Linux (ApplImage)

Linux .ApplImage バイナリを使用するパッケージを確認するには、まず Linux タブのステップ 1~3 を完了してから、次の手順を実行します。

1. GitHub の ApplImageUpdate [ページ](#) から、`validate-x86_64.AppImage` ファイルをダウンロードします。
2. ファイルをダウンロードした後、実行権限を追加するには、次のコマンドを実行します。

```
chmod a+x ./validate-x86_64.AppImage
```

3. 実行アクセス許可を追加するには、次のコマンドを実行します。

```
chmod a+x ./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
```

4. Deadline Cloud モニターの署名を確認するには、次のコマンドを実行します。

```
./validate-x86_64.AppImage ./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
```

出力に というフレーズが含まれている場合 `Validation successful`、署名が正常に検証され、Deadline Cloud モニターのインストールスクリプトを安全に実行できることを意味します。

Linux (Debian)

Linux .deb バイナリを使用するパッケージを確認するには、まず Linux タブのステップ 1~3 を完了します。

dpkg は、ほとんどの debian ベースの Linux ディストリビューションのコアパッケージ管理ツールです。ツールを使用して .deb ファイルを検証できます。

1. Deadline Cloud Monitor .deb ファイルをダウンロードします。

[Deadline Cloud Monitor のダウンロード \(.deb\)](#)

2. .deb ファイルを確認します。

```
dpkg-sig --verify deadline-cloud-monitor_amd64.deb
```

3. 出力は次のようになります。

```
Processing deadline-cloud-monitor_amd64.deb...
GOODSIG _gpgbuilder B840C08C29A90796A071FAA5F6CD3C 171200
```

4. .deb ファイルを検証するには、GOODSIG が出力に存在することを確認します。

Linux (RPM)

Linux .rpm バイナリを使用するパッケージを確認するには、まず Linux タブのステップ 1~3 を完了します。

1. Deadline Cloud Monitor .rpm ファイルをダウンロードします。

[Deadline Cloud Monitor \(.rpm\) をダウンロードする](#)

2. .rpm ファイルを確認します。

```
gpg --export --armor "Deadline Cloud" > key.pub
sudo rpm --import key.pub
rpm -K deadline-cloud-monitor.x86_64.rpm
```

3. 出力は次のようになります。

```
deadline-cloud-monitor.x86_64.rpm: digests signatures OK
```

4. .rpm ファイルを検証するには、`digests signatures OK`が出力にあることを確認します。

Deadline クラウドアシスタント

Deadline Cloud Assistant は、Deadline Cloud モニターに組み込まれた AI を活用したトラブルシューティングツールです。を使用して生成 AI を使用し、ジョブ設定、タスクステータス、セッションログ、CloudWatch データを分析して、レンダリングジョブの失敗を診断するのに役立ちます。アシスタントはブラウザで実行され、実用的な推奨事項を含むインテリジェントな根本原因分析を提供します。

Important

Deadline Cloud Assistant は生成 AI を搭載しています。AI モデルはレスポンスを生成し、レスポンスが不正確、不完全、または古い可能性があります。すべての推奨事項を確認してから対処してください。この機能の使用には、[AWS サービス条件](#)と[AWS 責任ある AI ポリシー](#)が適用されます。

トピック

- [アシスタントの仕組み](#)
- [重要な考慮事項](#)
- [Deadline Cloud Assistant の有効化](#)
- [必要なアクセス許可](#)
- [セキュリティ](#)
- [コスト](#)
- [サービスクォータとスロットリング](#)
- [トラブルシューティング](#)
- [その他のリソース](#)

アシスタントの仕組み

アシスタントを操作すると、レンダリングジョブの問題の理由として基盤モデルが使用されます。アシスタントは Deadline Cloud リソースと CloudWatch ログへの読み取り専用アクセス権を持ち、構造化されたトラブルシューティングワークフローに従います。

1. ジョブ設定とライフサイクルステータスを分析します

2. 失敗したタスクを識別し、障害パターンを調べる
3. セッション情報とセッションアクションの詳細を取得します。
4. CloudWatch ログでエラーパターンを分析する
5. 特定の推奨事項を含む根本原因分析を提供します

アシスタントは、以下のタスクもサポートします。

- 現在のページのログの概要
- モニター内の関連リソース (ワーカー、ログ、タスク) に移動する
- Deadline Cloud の概念と用語に関する質問に回答する
- レンダラー固有の問題のトラブルシューティング

すべての推論は、独自のサービスクォータ AWS アカウント を使用して 内で行われます。ページを更新すると、会話履歴がクリアされます。

重要な考慮事項

Deadline Cloud Assistant を使用する前に、AI 生成コンテンツとデータ処理に関する以下の考慮事項に注意してください。

- AI 生成レスポンス – アシスタントは生成 AI を使用してレスポンスを生成します。生成 AI と同様に、レスポンスは不正確、不完全、または古くなる可能性があります。環境を変更する前に、必ずレコメンデーションを確認してください。
- コンテンツの保護 — アシスタントの指示とツールは、デプロイプロセスの一環として不正または危険なコンテンツがないかスキャンされます。ただし、アシスタントは出力にランタイムコンテンツフィルタリングを適用しません。レスポンスには、予期しないコンテンツや不適切なコンテンツが含まれている場合があります。
- フィードバック – サムズアップアイコンとサムズダウンアイコンを使用して、個々のレスポンスに関するフィードバックを提供できます。一般的なフィードバックフォームは、アシスタントパネルからも入手できます (非 EU および非英国リージョンのみ)。 [AWS サポート](#) に連絡して問題を報告することもできます。

Deadline Cloud Assistant の有効化

Deadline Cloud 管理者のみがアシスタントを有効または無効にできます。アシスタントはデフォルトで無効になっています。

前提条件

アシスタントを使用するには、モニターに次のものがが必要です。

- Deadline Cloud コンソールを介してアシスタントを有効にする管理者
- モニターユーザーロールにアタッチされた IAM ポリシー

アシスタントの有効化

以下の手順を使用して、すべてのモニターユーザーのアシスタントを有効にします。

アシスタントを有効にするには

1. [Deadline Cloud コンソール](#)を開きます。
2. ナビゲーションペインで、モニターを選択し、モニターを選択します。
3. **編集** を選択してモニター設定を開きます。
4. Deadline Cloud Assistant を有効にするチェックボックスをオンにします。
5. **[保存]** を選択します。

アシスタントを有効にすると、Deadline Cloud は次のようになります。

- リージョン間の推論プロファイルにスコープされた `bedrock:InvokeModelWithResponseStream` アクセス許可を付与するモニターユーザーロールに IAM ポリシーを作成します。
- モニター設定 API を通じて有効状態を維持します。

アシスタントの無効化

アシスタントを無効にするには、次の手順に従います。

アシスタントを無効にするには

1. [Deadline Cloud コンソール](#)を開きます。
2. ナビゲーションペインで、モニターを選択し、モニターを選択します。
3. **編集** を選択してモニター設定を開きます。
4. Deadline Cloud Assistant を有効にするチェックボックスをオフにします。
5. **[保存]** を選択します。

必要なアクセス許可

モニターユーザーには、アシスタントを使用するための `bedrock:InvokeModelWithResponseStream` アクセス許可が必要です。管理者がアシスタントを有効にすると、Deadline Cloud は必要な IAM ポリシーをモニターユーザーロールに自動的にアタッチします。

Deadline Cloud IAM アクセス許可の詳細については、[「Deadline Cloud のアイデンティティベースのポリシーの例」](#)を参照してください。

IAM ポリシー

管理者がアシスタントを有効にすると、次の IAM ポリシーがモニターユーザーロールにアタッチされます。このポリシーは、モニターの地理的リージョンを対象とするクロスリージョン推論プロファイルを介してモデルを呼び出すアクセス許可を付与します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GrantCrisInferenceProfileAccess",
      "Effect": "Allow",
      "Action": "bedrock:InvokeModelWithResponseStream",
      "Resource": "arn:aws:bedrock:Region:AccountId:inference-profile/InferenceProfilePrefix.*",
      "Condition": {
        "StringEquals": {
          "aws:RequestedRegion": RequestedRegions
        }
      }
    }
  ],
}
```

```
{
  "Sid": "GrantCrisModelAccess",
  "Effect": "Allow",
  "Action": "bedrock:InvokeModelWithResponseStream",
  "Resource": "arn:aws:bedrock:*::foundation-model/*",
  "Condition": {
    "StringEquals": {
      "aws:RequestedRegion": RequestedRegions
    },
    "ArnLike": {
      "bedrock:InferenceProfileArn":
        "arn:aws:bedrock:Region:AccountId:inference-profile/InferenceProfilePrefix.*"
    }
  }
}
```

ポリシーは *のみ* を付与 `bedrock:InvokeModelWithResponseStream` します。他のアクションは許可されません。

クロスリージョン推論

アシスタントは、[クロスリージョン推論](#)を使用してモデルの可用性とスループットを最適化します。アシスタントを呼び出すと、は推論リクエストを処理するために、同じ地理的領域 AWS リージョン内の別の にリクエストをルーティングすることがあります。

- リクエストは、モニターのリージョンによって決定される地理的境界 AWS リージョン 内でルーティングされます。
- リージョン間で送信されるすべてのデータは AWS ネットワーク上に残り、パブリックインターネットを経由しません。
- データは 間で転送中に暗号化されます AWS リージョン。
- クロスリージョン推論には追加のルーティングコストはかかりません。料金は、推論プロファイルを呼び出すリージョンに基づいています。
- クロスリージョン推論リクエストは、ソースリージョンの CloudTrail に記録されます。 `additionalEventData.inferenceRegion` フィールドは、リクエストが処理された場所を識別します。

次の表は、モニターのリージョンに基づいて使用される地理的推論プロファイルと送信先リージョンを示しています。

クロスリージョン推論プロファイルマッピング

リージョンのモニターリング	推論プロファイルプレフィックス	送信先リージョン
us-east-1	us	us-east-1、us-east-2、us-west-2
us-east-2	us	us-east-1、us-east-2、us-west-2
us-west-2	us	us-east-1、us-east-2、us-west-2
eu-central-1	eu	eu-central-1、eu-north-1、eu-south-1、eu-south-2、eu-west-1、eu-west-3
eu-west-1	eu	eu-central-1、eu-north-1、eu-south-1、eu-south-2、eu-west-1、eu-west-3
eu-west-2	eu	eu-central-1、eu-north-1、eu-south-1、eu-south-2、eu-west-1、eu-west-2、eu-west-3
ap-northeast-1	jp	ap-northeast-1、ap-northeast-3
ap-southeast-2	au	ap-southeast-2、ap-southeast-4
ap-northeast-2	global	ap-northeast-2
ap-southeast-1	global	ap-southeast-1

global 推論プロファイルプレフィックスを使用するリージョンの場合、はサポートされている AWS リージョン 世界中の商用 にリクエストをルーティングすることがあります。

セキュリティ

Deadline Cloud Assistant は、既存の Deadline Cloud セキュリティモデル内で動作します。

- 読み取り専用アクセス – アシスタントは、Deadline Cloud リソースと CloudWatch ログに対してのみ読み取りオペレーション (取得、一覧表示、検索) を実行します。リソースを変更することはできません。
- カスタマーアカウント実行 — すべてのモデル呼び出しは、認証情報とサービスクォータ AWS アカウント を使用して で行われます。
- スコープ付きアクセス許可 – ポリシーは、地理的リージョンのクロスリージョン推論プロファイルにスコープされます。Monitor ユーザーは、 以外のアクションにアクセスできません `InvokeModelWithResponseStream`。
- セッションの分離 – 会話は個々のブラウザセッションに分離され、永続化または共有されません。
- 失敗クローズ – アシスタントが有効化されているかどうかを判断できない場合 (たとえば、`GetMonitorSettings`呼び出しが失敗した場合)、アシスタント UI は表示されません。
- 管理者コントロール – アシスタントを有効または無効にできるのは管理者のみです。モニターユーザーはアクセスを自己エスカレーションできません。
- 不正検出 – 不正検出機能はアシスタントの使用に適用されます。詳細については、「ユーザーガイド」の [「不正検出」](#) を参照してください。

モデル情報

Deadline Cloud Assistant は、クロスリージョン推論プロファイルを介してアクセスされる Anthropic Claude Sonnet 4.5 (`anthropic.claude-sonnet-4-5-20250929-v1:0`) を基盤モデルとして使用します。アシスタントには、一般的なデジタルコンテンツ作成アプリケーションの Deadline Cloud の公開ドキュメント、公開 AWS ドキュメント、公開ドキュメントから構築されたナレッジベースも含まれています。このナレッジベースは、呼び出し時にアシスタントによって取得されます。AWS は、どの Deadline Cloud アカウントの顧客データを使用してもアシスタントを構築または微調整しません。

データプライバシー

Deadline Cloud Assistant にはデータ保護ポリシーが適用されます。データ保護の詳細については、「ユーザーガイド」の [「データ保護」](#) を参照してください。

アシスタントは会話履歴をブラウザメモリにのみ保持します。ページを更新または閉じると、会話は完全に削除されます。アシスタントは、会話データをディスク、データベース、または AWS サービスに保持しません。

アカウントでモデル呼び出しログ記録が有効になっている場合、アシスタントの会話 (モデルに送信されたログコンテンツを含む) は、設定されたログ記録先 (Amazon S3 バケットまたは CloudWatch Logs ロググループ) にキャプチャされます。モデル呼び出しのログ記録はデフォルトで無効になっており、完全にユーザーの管理下にあります。詳細については、「ユーザーガイド」の [「モデル呼び出しログ記録」](#) を参照してください。

ネットワークパス

Deadline Cloud アシスタントは、Deadline Cloud モニターアプリケーションの一部としてブラウザで実行されます。アシスタントを操作すると、ブラウザはモニターユーザーの認証情報を使用してサービスエンドポイントに直接 API コール (InvokeModelWithResponseStream) を実行します。これらの呼び出しは、HTTPS (TLS 1.2 以降) 経由でリージョンのパブリックエンドポイントに送信されます。

アシスタントはブラウザで実行されるため、Amazon VPC インターフェイスエンドポイント (AWS PrivateLink) はアシスタントトラフィックには適用されません。PrivateLink のサポートは、ブラウザベースのアプリケーションではなく、Amazon VPC 内で実行されるサーバー側のワークロード向けに設計されています。

組織レベルのコントロール

モニターごとの管理トグルに加えて、AWS Organizations (Organizations) サービスコントロールポリシー (SCPs) を使用して、アシスタントを組織全体で制御できます。拒否する SCP は、モニター管理者がこの機能を有効にしても、アシスタントが機能する `bedrock:InvokeModelWithResponseStream` のを防ぎます。

次の SCP の例では、すべてのモデル呼び出しを拒否します。これにより、ポリシーがアタッチされている組織または組織単位 (OU) 内のすべてのアカウントでアシスタントが無効になります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyBedrockInvocations",
      "Effect": "Deny",
      "Action": "bedrock:InvokeModelWithResponseStream",
      "Resource": "*"
    }
  ]
}
```

SCPs 「Organizations ユーザーガイド」の [「サービスコントロールポリシー」](#) を参照してください。

Note

この SCP は、Deadline Cloud に関連しないものを含め、影響を受けるアカウントのすべてのモデル呼び出しをブロックします。アシスタントのみをブロックするには、代わりにモニター設定を使用して無効にします。

監査証跡

アシスタントのアクティビティは、AWS CloudTrail (CloudTrail) を通じて監査可能です。

- 呼び出し – CloudTrail は、各 `InvokeModelWithResponseStream` 呼び出しを管理イベントとしてログに記録します。ログエントリには、モデル ID、ユーザー ID、タイムスタンプ、ソース IP が記録されます。 `additionalEventData.inferenceRegion` フィールドは、リクエストが処理された場所を識別します。CloudTrail イベントには、プロンプトまたはレスポンスコンテンツは含まれません。
- Deadline Cloud リソースの読み取り – Deadline Cloud リソース (`GetJob`、`ListTasks`、など `SearchTasks`) に対するアシスタントの読み取りオペレーションは `ListSessions`、標準の Deadline Cloud API コールとして CloudTrail に記録されます。これらのログをクエリして、会話中にアシスタントがアクセスした特定のジョブ、タスク、セッションを特定できます。
- CloudWatch Logs の読み取り – アシスタントは、キューロール (`deadline:AssumeQueueRoleForRead`) またはフリートロール (`deadline:AssumeFleetRoleForRead`) を引き受けてワーカーログとタスクログを読み取ります。これらのロールの引き受けイベントは CloudTrail に記録されます。

不正検出

自動不正使用検出メカニズムは、すべてのアシスタントの使用に適用されます。詳細については、「ユーザーガイド」の [「悪用検出」](#) を参照してください。

フィードバックデータ

アシスタントは 2 つのフィードバックメカニズムを提供します。各メカニズムは異なるデータを送信します。

- サムアップ/ダウンボタン – アシスタントレスポンスのサムアップまたはサムダウンアイコンをクリックすると、感情インジケータ (正または負) とセッション識別子のみがテレメトリイベントとして記録されます。フィードバックイベントには、会話コンテンツ、ログデータ、プロンプトは含まれません。
- 一般的なフィードバックフォーム (非 EU および非英国リージョンのみ) – 吹き出しアイコンを使用して一般的なフィードバックを送信すると、フォームは明示的に入力した情報のみを送信します。これには、カテゴリの選択、件名、説明、オプションの E メールアドレスが含まれます。フォームには、モニターのリージョンと現在のページパスもメタデータとして含まれます。フォームフィールドに手動で入力しない限り、会話コンテンツやログデータは含まれません。一般的なフィードバックは AWS フィードバックサービスに送信されます。

データレジデンシー要件のため、EU および英国リージョンでは一般的なフィードバックは利用できません。テレメトリイベントには顧客コンテンツが含まれていないため、親指の上下フィードバックはすべてのリージョンで利用できます。

コスト

Deadline Cloud Assistant では、 の使用コストが発生します AWS アカウント。コストは、各インタラクション中に処理された入出力トークンの数に基づいています。アシスタントはクロスリージョン推論を使用するため、料金はモニターのソースリージョンに基づいて計算されます。

現在の料金については、「[の料金](#)」を参照してください。

Tip

アシスタントは、インタラクション中にコンテキストウィンドウの使用状況を表示し、トークンの消費をモニタリングするのに役立ちます。

アシスタントのコストの追跡

は、AWS Identity and Access Management (IAM) プリンシパルによるコスト配分をサポートします。このプリンシパルを使用して、チーム、プロジェクト、またはコストセンター全体のアシスタント推論コストを追跡および属性付けできます。アシスタントのコスト追跡を設定するには、モニターユーザーロールに組織構造を表す属性をタグ付けし、コスト配分のためにそれらのタグをアクティブ化します。

この手順を完了するには、ロールにタグを付け、請求コンソールにアクセスするための IAM アクセス許可が必要です。

アシスタントのコスト追跡を設定するには

1. モニターユーザーが引き受ける IAM ロールにタグを付けます。タグは、IAM コンソールまたはを使用して追加できます AWS CLI。次の例では、AWS CLIを使用します。

```
aws iam tag-role \  
  --role-name YourMonitorUserRole \  
  --tags Key=team,Value=YourTeam Key=project,Value=YourProject
```

team、projectなど、組織構造に合ったタグキーを選択しますcost-center。

2. コスト配分のタグをアクティブ化します。
 - a. [請求コンソール](#)を開きます。
 - b. ナビゲーションペインの Cost Organization で、Cost Allocation Tags を選択します。
 - c. IAM プリンシパルタグを検索して選択し、アクティブ化を選択します。
3. () では、アクティブ化された IAM プリンシパルタグでフィルタリングまたはグループ化してコストを表示します。
 - a. グループ化ディメンションとしてタグを選択します。
 - b. タグキーを選択します。

IAM ロールにタグを適用すると、タグキーがコスト配分タグページに表示されるまでに最大 24 時間かかることがあります。その後、タグがアクティブ化されるまでに最大 24 時間かかることがあります。

明細項目のコスト属性については、AWS Cost and Usage Report () データエクスポートを作成し、発信者 ID (IAM プリンシパル) 割り当てデータを含めるを選択します。エクスポートには、各リクエストの IAM ARN と、プレフィックスが付いた IAM プリンシパルタグを記録するline_item_iam_principal列が含まれますiamPrincipal/。

詳細については、AWS 「Billing User Guide」の「[Using IAM principal for cost allocation](#)」を参照してください。

サービスクォータとスロットリング

アシスタントは、アカウントのサービスクォータの対象となるオンデマンド推論を使用します。2つの主な制約は次のとおりです。

- 1分あたりのリクエスト数 (RPM) — 1分あたりに実行できるモデル呼び出しリクエストの数。
- 1分あたりのトークン (TPM) — 1分あたりに処理された入出力トークンの合計数。

デフォルトのクォータはリージョンによって異なります。一部のリージョンでは、デフォルトの制限が低く (最低 20 RPM)、アシスタントの使用が多いとスロットリングが発生する可能性があります。

クォータ引き上げのリクエスト

アシスタントの使用時にスロットリングエラーが発生した場合は、サービスクォータの引き上げをリクエストできます。

クォータの増加をリクエストするには

1. [Service Quotas コンソール](#) を開きます。
2. ナビゲーションペインで、AWS サービスを選択し、 を選択します。
3. アシスタントが使用するモデルのクォータを見つけます (関連するモデルの に関連するクォータを探InvokeModelWithResponseStreamします)。
4. クォータ名を選択し、アカウントレベルで引き上げをリクエストを選択します。
5. 目的のクォータ値を入力し、リクエストを送信します。

詳細については、「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

Note

リージョンがクロスリージョン推論を使用している場合は、送信先リージョンのサービスクォータも適用されます。クロスリージョン推論プロファイルは、最小 200 RPM をサポートしているため、単一リージョンの制限が低いリージョンでのスロットリングを軽減できます。

クォータ使用状況のモニタリング

CloudWatch メトリクスを使用してクォータの使用状況をモニタリングできます。スロットリングメトリクスに CloudWatch アラームを設定して、クォータ制限に近づいているタイミングを事前に特定します。詳細については、「ユーザーガイド」の「[Monitoring](#)」を参照してください。

トラブルシューティング

このセクションでは、Deadline Cloud Assistant を使用する際に発生する可能性がある一般的な問題とその解決方法について説明します。

Assistant がエラーを返すか、応答が遅い

アシスタントはエラーメッセージを表示するか、レスポンスの生成に時間がかかります。

一般的な原因

エラーや応答の遅延は通常、サービスクォータを超えたとき、またはモニタユーザーロールに必要なアクセス許可がない場合に発生します。

解像度

- リージョンのサービスクォータを確認します。クォータ[サービスクォータとスロットリング](#)のチェックと引き上げの詳細については、「」を参照してください。
- スロットリングエラーが発生した場合は、リージョンでのサービスクォータの引き上げをリクエストします。
- モニターユーザーロールに必要なアクセス許可が にリストされていることを確認します [必要なアクセス許可](#)。

Assistant が Amazon CloudWatch logs にアクセスできない

アシスタントは CloudWatch からログデータを取得または分析できません。

一般的な原因

キューロールまたはフリートロールに必要な CloudWatch Logs 読み取りアクセス許可がない場合、アシスタントはログを取得できません。

解像度

アシスタントは、タスクログを読み取るためにキューロール (を使用 `deadline:AssumeQueueRoleForRead`) を引き受け、ワーカーログを読み取るためにフリートロール (を使用 `deadline:AssumeFleetRoleForRead`) を引き受けます。これらのロールに CloudWatch Logs の読み取りアクセス許可があることを確認します。

その他のリソース

次のリソースは、Deadline Cloud Assistant に関する追加情報を提供します。

- [とは](#) ユーザーガイドの
- ユーザーガイドの [データ保護](#)
- 「ユーザーガイド」の [「不正検出」](#)
- ユーザーガイドの [モデル呼び出しログ記録](#)
- 「ユーザーガイド」の [「クロスリージョン推論」](#)
- ユーザーガイドの [CloudTrail を使用して API コールをモニタリングする](#)
- [料金表](#)
- AWS 請求ユーザーガイドの [コスト配分に IAM プリンシパルを使用する](#)
- Organizations ユーザーガイドの [サービスコントロールポリシー](#)
- [Deadline Cloud のアイデンティティベースのポリシーの例](#)
- [AWS 責任ある AI ポリシー](#)
- [AWS データプライバシーに関するよくある質問](#)

AWS Deadline Cloud のモニタリング

モニタリングは、AWS Deadline Cloud (Deadline Cloud) と AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。マルチポイント障害が発生した場合は、その障害をより簡単にデバッグできるように、AWS ソリューションのすべての部分からモニタリングデータを収集します。Deadline Cloud のモニタリングを開始する前に、以下の質問に対する回答を含むモニタリング計画を作成する必要があります。

- どのような目的でモニタリングしますか？
- どのリソースをモニタリングしますか？
- どのくらいの頻度でこれらのリソースをモニタリングしますか？
- どのモニタリングツールを使用しますか？
- 誰がモニタリングタスクを実行しますか？
- 問題が発生したときに誰が通知を受け取りますか？

AWS および Deadline Cloud には、リソースをモニタリングし、潜在的なインシデントに対応するために使用できるツールが用意されています。これらのツールの中には、モニタリングを行うものもあれば、手動による介入を必要とするものもあります。モニタリングタスクはできるだけ自動化する必要があります。

- Amazon CloudWatch は、AWS リソースと AWS で実行しているアプリケーションをリアルタイムでモニタリングします。メトリクスの収集と追跡、カスタマイズしたダッシュボードの作成、および指定したメトリクスが指定したしきい値に達したときに通知またはアクションを実行するアラームの設定を行うことができます。例えば、CloudWatch で Amazon EC2 インスタンスの CPU 使用率などのメトリクスを追跡し、必要に応じて新しいインスタンスを自動的に起動できます。詳細については、「[Amazon CloudWatch ユーザーガイド](#)」を参照してください。

Deadline Cloud には 3 つの CloudWatch メトリクスがあります。

- Amazon CloudWatch Logs では、Amazon EC2 インスタンス、CloudTrail、およびその他のソースからのログファイルをモニタリング、保存、およびアクセスできます。CloudWatch Logs は、ログファイル内の情報をモニタリングし、特定のしきい値が満たされたときに通知します。高い耐久性を備えたストレージにログデータをアーカイブすることも可能です。詳細については、「[Amazon CloudWatch Logs ユーザーガイド](#)」を参照してください。
- Amazon EventBridge を使用すると、AWS サービスを自動化し、アプリケーションの可用性の問題やリソースの変更などのシステムイベントに自動的に対応できます。AWS サービスからのイベ

ントは、ほぼリアルタイムで EventBridge に配信されます。簡単なルールを記述して、注目するイベントと、イベントがルールに一致した場合に自動的に実行するアクションを指定できます。詳細については、「[Amazon EventBridge ユーザーガイド](#)」を参照してください。

- AWS CloudTrail は、AWS アカウントによって、またはアカウントに代わって行われた API コールおよび関連イベントをキャプチャし、指定した Amazon S3 バケットにログファイルを配信します。が呼び出したユーザーとアカウント AWS、呼び出し元のソース IP アドレス、および呼び出しの発生日時を特定できます。詳細については、[AWS CloudTrail ユーザーガイド](#)をご参照ください。

詳細については、Deadline Cloud デベロッパーガイドの以下のトピックを参照してください。

- [CloudTrail ログ](#)
- [EventBridge を使用したイベントの管理](#)
- [CloudWatch によるモニターリング](#)

のクォータ Deadline Cloud

AWS Deadline Cloud は、ジョブの処理に使用できるファーム、フリート、キューなどのリソースを提供します。を作成すると AWS アカウント、それぞれのリソースにデフォルトのクォータが設定されます AWS リージョン。

Service Quotas は、 のクォータを表示および管理できる中心的な場所です AWS のサービス。使用する多くのリソースのクォータの引き上げをリクエストすることもできます。

のクォータを表示するには Deadline Cloud、 [Service Quotas コンソール](#)を開きます。ナビゲーションペインで、[AWS のサービス] を選択し、次に [Deadline Cloud] を選択します。

クォータの引き上げをリクエストするには、「Service Quotas ユーザーガイド」の「[クォータ引き上げリクエスト](#)」を参照してください。Service Quotas でクォータがまだ利用できない場合は、[サービスクォータ引き上げフォーム](#)を使用します。

AWS アカウントには、次のクォータが関連しています Deadline Cloud。

次の表に、サービスマネージドフリートで使用される永続的ストレージボリュームのクォータを示します。永続的ストレージの詳細については、「」を参照してください [サービスマネージドフリートの永続的ストレージ](#)。

名前	デフォルト	引き上げ可能	説明
ファームあたりの関連メンバー数	サポートされている各リージョン: 75	はい	現在の AWS リージョンの各ファームに関連付けることができるメンバーの最大数。
フリートあたりの関連メンバー数	サポートされている各リージョン: 75	はい	現在の AWS リージョンの各フリートに関連付けることができるメンバーの最大数。

名前	デフォルト	引き上げ可能	説明
ジョブあたりの関連メンバー数	サポートされている各リージョン: 75	いいえ	現在の AWS リージョンの各ジョブに関連付けることができるメンバーの最大数。
キューあたりの関連メンバー数	サポートされている各リージョン: 75	いいえ	現在の AWS リージョンの各キューに関連付けることができるメンバーの最大数。
ファームあたりの予算数	サポートされている各リージョン: 20	<u>可</u> <u>能</u>	現在の AWS リージョンのファームあたりの予算の最大数
リージョンあたりのファーム数	サポートされている各リージョン: 2	<u>あ</u> <u>り</u>	現在の AWS リージョンで作成できるファームの最大数。
ファームあたりのフリート数	サポートされている各リージョン: 5	<u>あ</u> <u>り</u>	現在の AWS リージョンの各ファームに作成できるフリートの最大数。
ファームあたりのジョブ数	サポートされている各リージョン: 100,000	<u>あ</u> <u>り</u>	現在の AWS リージョンのファームあたりのジョブの最大数。
リージョンあたりのライセンスエンドポイント数	サポートされている各リージョン: 5	<u>あ</u> <u>り</u>	現在の AWS リージョンのライセンスエンドポイントの最大数。

名前	デフォルト	引き上げ可能	説明
ライセンスエンドポイントあたりのライセンスセッション数	サポートされている各リージョン: 500	あり	現在の AWS リージョンのライセンスエンドポイントあたりのライセンスセッションの最大数。
ファームあたりの制限数	サポートされている各リージョン: 50	可能	現在の AWS リージョンの各ファームに対して作成できる制限の最大数。
リージョンあたりのモニター数	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンのモニターの最大数。
リージョンあたりのオンデマンド G インスタンス GPU 数	サポートされている各リージョン: 1	あり	現在の AWS リージョンのすべてのサービスマネージドフリートにプロビジョニングできるオンデマンド G インスタンス GPU の最大数。
リージョンあたりのオンデマンド vCPU 数	サポートされている各リージョン: 50	可能	現在の AWS リージョンのすべてのサービスマネージドフリートにプロビジョニングできるオンデマンド vCPUs の最大数。
キューあたりのキュー環境数	サポートされている各リージョン: 10	いいえ	現在の AWS リージョンの各キューに対して作成できるキュー環境の最大数。

名前	デフォルト	引き上げ可能	説明
ファームあたりのキューフリート関連付け数	サポートされている各リージョン: 100	可能	現在の AWS リージョンのファームあたりのキューフリートの関連付けの最大数
キューあたりのキュー制限関連付け数	サポートされている各リージョン: 10	あり	現在の AWS リージョンの各キューに関連付けることができる制限の最大数。
ファームあたりのキュー数	サポートされている各リージョン: 20	可能	現在の AWS リージョンの各ファームに対して作成できるキューの最大数。
フリートあたりのリソース設定数	サポートされている各リージョン: 1	あり	各フリートに追加できる VPC Lattice リソース設定の最大数。
リージョンあたりのスポット G インスタンス GPU 数	サポートされている各リージョン: 1	あり	現在の AWS リージョンのすべてのサービスマネージドフリートにプロビジョニングできるスポット G インスタンス GPU の最大数。

名前	デフォルト	引き上げ可能	説明
リージョンあたりのスポット vCPU 数	サポートされている各リージョン: 500	あ り	現在の AWS リージョン内のすべてのサービスマネージドフリートにプロビジョニングできるスポット vCPUs の最大数。
ジョブあたりのステップ数	サポートされている各リージョン: 200	あ り	現在の AWS リージョンのジョブあたりのステップの最大数。
汎用 SSD (gp3) ボリュームのストレージ (TiB)	サポートされている各リージョン: 50	可 能	TiB で測定され、現在の AWS リージョン内のすべてのフリートで使用できる EBS ストレージの最大集約量。
永続汎用 SSD (gp3) ボリュームのストレージ、TiB 単位	サポートされている各リージョン: 50	可 能	現在の AWS リージョンの永続的な汎用 SSD (gp3) ボリュームにプロビジョニングできる TiB 単位のストレージの最大集約量。
ファームあたりのストレージプロファイル数	サポートされている各リージョン: 50	い い え	現在の AWS リージョンの各ファームに作成できるストレージプロファイルの最大数。

名前	デフォルト	引き上げ可能	説明
チャンクあたりのタスク数	サポートされている各リージョン: 150	いいえ	ジョブの送信時に 1 つのチャンクに結合できるタスクの最大数。
ジョブあたりのタスク数	サポートされている各リージョン: 10,000	あり	現在の AWS リージョンのジョブあたりのタスクの最大数。
ステップあたりのタスク数	サポートされている各リージョン: 10,000	あり	現在の AWS リージョンのステップあたりのタスクの最大数。
リージョンあたりの wait-and-save vCPU 数	サポートされている各リージョン: 50	可能	現在の AWS リージョンのすべてのサービスマネージドフリートにプロビジョニングできる wait-and-save vCPUs の最大数。
ファームあたりのワーカー数	サポートされている各リージョン: 7,500	あり	現在の AWS リージョンのファームあたりのワーカーの最大数。

を使用した Deadline Cloud AWS リソースの作成 AWS CloudFormation

AWS Deadline Cloud は と統合されています。これは AWS CloudFormation、AWS リソースとインフラストラクチャの作成と管理に費やす時間を短縮できるように、リソースのモデル化とセットアップに役立つサービスです。必要なすべての AWS リソース (ファーム、キュー、フリートなど) を記述するテンプレートを作成し、それらのリソースを CloudFormation プロビジョニングして設定します。

を使用すると CloudFormation、テンプレートを再利用して Deadline Cloud リソースを一貫して繰り返しセットアップできます。リソースを一度記述し、複数の AWS アカウント およびリージョンで同じリソースを何度もプロビジョニングします。

Deadline Cloud と CloudFormation テンプレート

Deadline Cloud および関連サービスのリソースをプロビジョニングして設定するには、[CloudFormation テンプレート](#)を理解する必要があります。テンプレートは、JSON や YAML でフォーマットされたテキストファイルです。これらのテンプレートは、CloudFormation スタックでプロビジョニングするリソースを記述します。JSON または YAML に慣れていない場合は、デザイナーを使用して CloudFormation CloudFormation テンプレートの使用を開始できます。詳細については、「AWS CloudFormation ユーザーガイド」の「[CloudFormation Designer とは](#)」を参照してください。

Deadline Cloud は、でのファーム、キュー、フリートの作成をサポートしています CloudFormation。ファーム、キュー、フリートの JSON テンプレートと YAML テンプレートの例を含む詳細については、「AWS CloudFormation ユーザーガイド」の[AWS 「Deadline Cloud」](#)を参照してください。

の詳細 CloudFormation

詳細については CloudFormation、次のリソースを参照してください。

- [AWS CloudFormation](#)
- [AWS CloudFormation ユーザーガイド](#)
- [CloudFormation API リファレンス](#)
- [AWS CloudFormation コマンドラインインターフェイスユーザーガイド](#)

トラブルシューティング

次の手順とヒントは、Deadline Cloud AWS ファームとリソースに関する問題のトラブルシューティングに役立ちます。

トピック

- [ユーザーがファーム、フリート、またはキューを表示できないのはなぜですか？](#)
- [ワーカーがジョブを取得しないのはなぜですか？](#)
- [ワーカーが停止しているのはなぜですか？](#)
- [Deadline Cloud ジョブのトラブルシューティング](#)
- [Deadline Cloud モニターデスクトップアプリケーションログ](#)
- [その他のリソース](#)

ユーザーがファーム、フリート、またはキューを表示できないのはなぜですか？

ユーザーアクセス

Deadline Cloud モニターにファーム、フリート、またはキューが表示されない場合、ファームとリソースへのアクセスに問題がある可能性があります。

ファームにアクセスできないユーザーは、Deadline Cloud モニターで「利用可能なファームはありません」というメッセージを受け取ります。

ファーム、フリート、またはキューに正しいユーザーまたはグループが割り当てられていることを確認するには

1. AWS Deadline Cloud コンソールで、ファーム、フリート、またはキューを検索し、アクセス管理を選択します。
2. グループタブはデフォルトで選択されています。グループごとにアクセス許可を割り当てる場合は、グループがリストに表示され、アクセスレベルが割り当てられます。

グループがリストにない場合は、グループの追加を選択して、グループに許可を割り当てます。

3. ユーザーごとにアクセス許可を割り当てる場合は、ユーザータブを選択します。ユーザーがリストに表示され、アクセスレベルが割り当てられている必要があります。

ユーザーがリストにない場合は、ユーザーを追加を選択してユーザーに許可を割り当てます。

ユーザーがグループに割り当てられていることを確認するには

1. AWS Deadline Cloud コンソールで、ファーム、フリート、またはキューを検索し、アクセス管理を選択します。
2. グループタブはデフォルトで選択されています。メンバーを表示するグループ名を選択します。
3. ユーザーがグループにリストされていない場合は、追加する必要があります。

デフォルトの ID 設定を使用している場合は、Identity Center コンソールでユーザーをグループに直接追加できます。Okta や などの外部 ID プロバイダーに接続されている場合は Google Workspace、ID プロバイダーのグループにユーザーを追加できます。

Note

一部の外部 ID プロバイダーはユーザーを同期しますが、グループは Identity Center に同期しません。この場合、グループではなくユーザーに直接アクセス許可を割り当てることを検討してください。

Deadline Cloud へのユーザーアクセスの管理の詳細については、「」を参照してください [Deadline Cloud でのユーザーの管理](#)。

ワーカーがジョブを取得しないのはなぜですか？

フリートロールの設定

ワーカーが作成されても初期化が完了せず、ジョブの処理を開始しない場合、フリートロールが正しく設定されていないことが原因です。

これが起こっていることを確認するには、アクセス拒否エラーがないか CloudTrail ログを確認します。アクセス拒否の問題を確認したら、フリートに移動し、ロール設定を正しいアクセス許可に更新します。詳細については、Deadline [CloudTrail ログ](#)」を参照してください。

ワーカーが停止しているのはなぜですか？

OpenJD 環境からのワーカーの停止

ワーカーは長時間実行されるenvExitセッションアクションで停止する可能性があります。これは、OpenJD テンプレートを上書きし、環境終了アクションのタイムアウトを 5 分以上に設定するジョブテンプレートを使用する場合に発生する可能性があります。Deadline Cloud モニターは、この状況でスタックしているワーカーをある程度可視化しますが、関連するキューで利用可能な作業とRUNNINGワーカーを相互参照する必要があります。

スタックしたワーカーを見つけるには、Deadline Cloud モニターのすべてのフリートを確認し、次の手順を実行します。

1. ワーカーステータス列で、RUNNINGワーカーを検索します。
2. フリートの詳細セクションから、関連する各キューに移動します。
3. 関連付けられた各キューで、RUNNING、READYまたはPENDINGのジョブを検索します。関連付けられたすべてのキューにそれらの状態のジョブがない場合、ワーカーは環境の終了を実行しています。

この状態でワーカーがスタックするのを停止するには、次の AWS CLI コマンドを使用します。

```
aws deadline update-worker \  
  --farm-id $FARM_ID \  
  --fleet-id $FLEET_ID \  
  --worker-id $WORKER_ID \  
  --status STOPPED
```

コマンドを実行すると、プログラムが終了するとワーカーエージェントは再起動します。その後、ワーカーはオンラインに戻り、関連付けられたキューからより多くのジョブを実行します。キューに環境終了アクションのタイムアウトが 5 分を超えるジョブがさらに含まれている場合、ワーカーは再びスタックします。この場合、ワーカーが終了しなくなるまでこのプロセスを繰り返す必要があります。

この問題を回避するには、ジョブテンプレートを使用するときにタイムアウトオプションを 5 分以内に設定します。

Deadline Cloud ジョブのトラブルシューティング

AWS Deadline Cloud のジョブに関する一般的な問題については、以下のトピックを参照してください。

ジョブの作成が失敗したのはなぜですか？

クォータの検証

ジョブが検証チェックに失敗する理由には、次のようなものがあります。

- ジョブテンプレートが OpenJD 仕様に従っていない。
- ジョブに含まれるステップが多すぎます。
- ジョブの合計タスクが多すぎます。
- ジョブの作成を妨げる内部サービスエラーが発生しました。

ジョブ内のステップとタスクの最大数のクォータを表示するには、Service Quotas コンソールを使用します。詳細については、「[のクォータ Deadline Cloud](#)」を参照してください。

CHUNK[INT] タスクパラメータエラー

ジョブの作成に失敗し、次のエラーメッセージが表示された場合は、TASK_CHUNKING 拡張機能をジョブテンプレートに追加する必要があります。

```
The CHUNK[INT] task parameter requires the TASK_CHUNKING extension.
```

この問題を解決するには、ジョブテンプレートに以下を追加します。

```
extensions:  
- TASK_CHUNKING
```

ジョブに互換性がないのはなぜですか？

ジョブがキューと互換性がない一般的な理由は次のとおりです。

- ジョブが送信されたキューに関連付けられているフリートはありません。Deadline Cloud モニターを開き、キューにフリートが関連付けられていることを確認します。キューを表示する方法

の詳細については、「」を参照してください[Deadline Cloud でキューとフリートの詳細を表示する](#)。

- ジョブには、キューに関連付けられているフリートによって満たされないホスト要件があります。確認するには、ジョブテンプレートのhostRequirementsエントリをファーム内のフリートの設定と比較します。いずれかのフリートがホスト要件を満たしていることを確認します。フリートの互換性の詳細については、「[フリートの互換性の確認](#)」を参照してください。フリート設定を表示するには、「」を参照してください[Deadline Cloud でキューとフリートの詳細を表示する](#)。

ジョブの準備が整うのはなぜですか？

ジョブが READY状態でスタックしているように見える理由には、次のようなものがあります。

- キューに関連付けられているフリートの最大ワーカー数は 0 に設定されています。確認するには、「」を参照してください[Deadline Cloud でキューとフリートの詳細を表示する](#)。
- キューには優先度の高いジョブがあります。確認するには、「」を参照してください[Deadline Cloud でキューとフリートの詳細を表示する](#)。
- カスタマーマネージドフリートの場合は、自動スケーリング設定を確認します。詳細については、「Deadline Cloud Developer Guide」の「Create [fleet infrastructure with an Amazon EC2 Auto Scaling group](#)」を参照してください。

ジョブが失敗したのはなぜですか？

ジョブは、さまざまな理由で失敗する可能性があります。問題を検索するには、Deadline Cloud モニターを開き、失敗したジョブを選択します。失敗したタスクを選択し、タスクのログを表示します。手順については、「[Deadline Cloud でセッションログとワーカーログを表示する](#)」を参照してください。

- ライセンスエラーが発生した場合、またはソフトウェアに有効なライセンスがないためにウォーカーマークが発生した場合は、ワーカーが必要なライセンスサーバーに接続できることを確認してください。詳細については、「Deadline Cloud Developer Guide」の「[Connect customer-managed fleets to a license endpoint](#)」を参照してください。
- 最後のセッションアクションメッセージまたはプロセス終了コードは、ジョブが失敗した理由に関する情報を提供する場合があります。を使用してWindowsいて、終了コードが負の場合は、終了コードの署名なしバージョンを検索してみてください。

```
2,147,483,647 - |your exit code|
```

ステップが保留になっているのはなぜですか？

ステップは、1 つ以上の依存関係が完了していない場合、PENDING状態のままになることがあります。Deadline Cloud モニターを使用して、依存関係の状態を確認できます。手順については、「[Deadline Cloud でステップを表示する](#)」を参照してください。

Deadline Cloud モニターデスクトップアプリケーションログ

Deadline Cloud モニターデスクトップアプリケーションは、クラッシュやその他の予期しない動作を調査するために使用できる診断ログを書き込みます。デスクトップアプリケーションに関する問題を報告するときは、診断に役立つ関連ログファイルを含めます。

ログファイルの場所は、オペレーティングシステムによって異なります。

Windows

```
%APPDATA%\com.amazonaws.deadline.monitor\logs
```

macOS

```
~/Library/Logs/com.amazonaws.deadline.monitor/
```

Linux

```
~/.config/com.amazonaws.deadline.monitor/logs
```

その他のリソース

追加情報とリソースは [GitHub](#) にあります。

Deadline Cloud リリースノート

このページには、AWS Deadline Cloud の最新リリースと更新に関する情報が含まれています。

日付	タイトル	説明
2026-06-03	Blender と Autodesk Maya のプラグイン同期	サービスマネージドフリートは、Blender と Autodesk Maya のプラグイン同期をサポートするようになりました。プラグインファイルをキューのジョブアタッチメント S3 バケットの所定のパスにアップロードします。Deadline Cloud は、ジョブセッションの開始時にワーカーに自動的に同期します。
2026-06-02	Blender 5.1 のサポートと LTS の更新	Deadline Cloud が Blender 5.1 をサポートするようになりました。Blender 5.1 から Deadline Cloud レンダーファームにレンダリングジョブと合成ジョブを送信できます。さらに、Blender 4.5 LTS が 4.5.5 から 4.5.10 に、Blender 4.2 LTS が 4.2.16 から 4.2.21 に更新され、アップストリームレンダリングの修正と安定性の向上が組み込まれています。
2026-05-28	サービスマネージドフリートの永続的ストレージ	サービスマネージドフリートで永続ストレージを設定して、ワーカーライフサイクルイベント全体でデータを保存できるようになりました。

日付	タイトル	説明
		<p>永続的ストレージは、ルートブートボリュームとは別に、専用の Amazon EBS gp3 ボリュームを使用します。これらのボリュームは、フリートがワーカーをリサイクルするときに、アプリケーションキャッシュ、conda パッケージのインストール、ワークスペースを維持します。詳細については、Deadline Cloud ユーザーガイドの「サービスマネージドフリートの永続的ストレージ」を参照してください。</p>
2026-05-28	SOC 1、2、3 のコンプライアンス	<p>Deadline Cloud が、System and Organization Controls (SOC) 1、2、3 のコンプライアンスに関する AWS Services in Scope ページに表示されるようになりました。サードパーティーの監査レポートは、AWS Artifact を使用してダウンロードできます。詳細については、「Deadline Cloud のコンプライアンス検証」を参照してください。</p>

日付	タイトル	説明
2026-05-18	サービスマネージドフリートでの conda キュー環境の高速セットアップ	<p>サービスマネージドフリートのデフォルトの conda キュー環境が改善されました。新しい conda-queue-env-enter コマンドと conda-queue-env-exit コマンドが追加されました。これにより、環境の作成が高速化され、繰り返し実行がさらに改善されます。詳細については、https://github.com/aws-deadline/deadline-cloud-samples/tree/mainline/queue_environments を参照してください。</p>
2026-05-14	Deadline Cloud Monitor 1.1.9	<p>Deadline Cloud Monitor デスクトップアプリケーションのこのリリースには、次の変更が含まれています。</p> <ul style="list-style-type: none">• Deadline クライアントをアップグレードして、ジョブアタッチメントの入力と出力のフィルタリングをサポートします。• ログイン時に一部のユーザーが最新バージョンに更新するよう求められないバグを修正しました。• Deadline Cloud Monitor がシステム依存関係がないために起動できない場合のエラーメッセージを改善しました。

日付	タイトル	説明
2026-05-11	Deadline Cloud Monitor のアプリ内リリースノート	Deadline Cloud Monitor で製品更新を直接参照できるようになりました。リリースノートページには、カテゴリフィルタリングと検索を含む新機能、修正、ドキュメントの更新が表示されるため、モニターを離れることなく関連する変更を見つけることができます。Monitor のサイドナビゲーションの下部にあるセカンダリナビゲーションメニューからリリースノートにアクセスします。
2026-04-29	NVIDIA RTX PRO Server 6000 GPU アクセラレーターのサポート	サービスマネージドフリートを作成するときに、RTX PRO Server 6000 GPU アクセラレーターを選択できるようになりました。このアクセラレーターは G7e インスタンスファミリーを使用し、アクセラレーターごとに 96 GiB の GPU メモリを提供します。このアクセラレーターを使用するには、フリートを設定するときに、アクセラレーター名に rtx-pro-server-6000、ドライバーバージョンに grid:r580 を選択します。

日付	タイトル	説明
2026-04-20	Deadline Cloud Monitor 1.1.8	<p>Deadline Cloud Monitor デスクトップアプリケーションのこのリリースには、次の変更が含まれています。</p> <ul style="list-style-type: none">• Deadline クライアントをアップグレードして、タスクをチャンクで操作できるようにします。• ランタイムの依存関係をバンドルして、ジョブの送信を高速化します。• 高速起動のためにロングスプラッシュアニメーションを削除します。• いくつかの小さなバグを修正しました。• 英語以外の言語の翻訳が改善されました。

日付	タイトル	説明
2026-04-17	<u>インテリジェントなジョブのトラブルシューティングのための Deadline Cloud Assistant</u>	Deadline Cloud Monitor に組み込まれた AI を活用したトラブルシューティングアシスタントである Deadline Cloud Assistant を使用して、レンダリングジョブの失敗を診断できるようになりました。アシスタントは、ジョブ設定、関連する Amazon CloudWatch logs、サポートされているデジタルコンテンツ作成アプリケーションのドキュメントを分析します。ブラウザで実行され、AWS アカウントで Amazon Bedrock を使用するため、データはセキュリティ境界内に留まります。開始するには、AWS マネジメントコンソールの Monitor 設定で Deadline Cloud Assistant を有効にするを選択します。
2026-04-10	<u>複数のリージョンでのモニター作成のサポート</u>	Deadline Cloud モニターを別のリージョンの IAM Identity Center インスタンスに接続できるようになりました。追加の IAM Identity Center 設定は必要ありません。Deadline Cloud は、インスタンスが配置されているリージョンから ID データを読み取ります。

日付	タイトル	説明
2026-04-06	新しいバッチ APIs—一括リソース操作を簡素化	<p>8 つの新しい Batch APIs を使用すると、1 回の API コールで複数のリソースを取得または更新できるため、一括オペレーションのオーバーヘッドを削減できます。新しい Batch Get APIs。</p> <ul style="list-style-type: none">• BatchGetJob• BatchGetSession• BatchGetSessionAction• BatchGetStep• BatchGetTask• BatchGetWorker <p>新しいバッチ更新 APIs は次のとおりです。</p> <ul style="list-style-type: none">• BatchUpdateJob• BatchUpdateTask
2026-04-03	Foundry Nuke 17 のサポート	<p>Deadline Cloud が Foundry Nuke 17 をサポートするようになりました。レンダリングジョブと合成ジョブを Nuke 17 から Deadline Cloud レンダーファームに送信できます。</p>

日付	タイトル	説明
2026-04-03	GPU フリートの NVIDIA GRID R580 サポート	Windows Server 2022 と Amazon Linux 2023 の両方で、Deadline Cloud のサービスマネージドフリートで NVIDIA Virtual GPU Software v19.0 (GRID R580) を使用できるようになりました。このソフトウェアバージョンを使用するには、GPU アクセラレーションのサービスマネージドフリートを作成するときに、ドライバーバージョンに <code>grid:r580</code> を選択します。
2026-04-02	Unreal Engine 5.7 のサポート	Deadline Cloud で Epic Games Unreal Engine 5.7 を使用できるようになりました。Windows からジョブを送信し、Windows のサービスマネージドフリートで <code>conda</code> パッケージを使用できます。
2026-04-02	バランスの取れたスケジューリングのサポート	各キューでスケジューリングを設定することで、Deadline Cloud がワーカーにジョブを分散する方法を制御できるようになりました。

日付	タイトル	説明
2026-03-30	フリートの自動スケーリング設定	Deadline Cloud は、3 つの新しいフリートの自動スケーリング設定をサポートするようになりました。スケールアウトレートを使用すると、ワーカーの起動速度を設定できます。ワーカーのアイドル時間を使用すると、ワーカーがシャットダウンするまでの待機時間を設定できます。スタンバイワーカー数を使用すると、ジョブをすばやく開始できるように、アイドルワーカーを準備しておくことができます。
2026-03-25	ファームのコストスケール係数	Usage Explorer と Budget Manager でコストをモデル化するように、ファームのコストスケール係数を設定できるようになりました。ファームのコスト計算に割引やプレミアムの適用して、Deadline Cloud の使用状況データを組織の実際のコストに合わせることができます。

日付	タイトル	説明
2026-03-23	送信者インストーラ v2026-03-23 更新 Maya 送信者	<p>新しい送信者インストーラは、次のコンポーネントを更新します。</p> <ul style="list-style-type: none">• Maya: 0.15.13 → 0.15.14 (リリースノート) <p>インストーラは Deadline Cloud クライアントの GUI 依存関係をバンドルし、インターネットアクセスなしで完全なインストールを可能にするようになりました。</p>
2026-03-16	進行中のジョブの推定残り時間	<p>進行中のジョブの推定残り時間を表示するには、<code>--show-estimates</code> フラグを期限ジョブリストと期限ジョブ取得コマンドとともに使用します。この見積みでは、平均タスク完了時間を使用し、人間が読み取れる期間として結果を表示します。</p>
2026-03-13	After Effects 25.6 および 26.0 のサポート	<p>Adobe After Effects バージョン 25.6 および 26.0 がサポートされるようになりました。送信者のサポートは Windows および macOS で利用でき、conda パッケージは Windows のサービスマネージドフリートで利用できます。</p>

日付	タイトル	説明
2026-03-11	送信者インストーラ v2026-03-11 がリリースされました	<p>次のコンポーネントを更新する新しい送信者インストーラがリリースされました。</p> <ul style="list-style-type: none">• 3ds-max: 0.1.9 → 0.1.10 (リリースノート)• 後続効果: 0.4.4 → 0.4.5 (リリースノート)• cinema-4d: 0.9.2 → 0.10.0 (リリースノート)• deadline-cloud: 0.54.1 → 0.54.2 (リリースノート)• houdini: 0.7.10 → 0.7.11 (リリースノート)
2026-03-10	Blender 5.0 のサポート	<p>Deadline Cloud は、Cycles、Eevee、Workbench を含むすべての組み込みレンダーエンジンで Blender 5.0 をサポートするようになりました。送信者のサポートは Windows、macOS、Linux で利用でき、conda パッケージは Linux のサービスマネージャドフリートで利用できます。</p>
2026-03-02	送信者インストーラ v2026-03-02 の更新 Blender とコアコンポーネント	<p>次のコンポーネントを更新する新しい送信者インストーラがリリースされました。</p> <ul style="list-style-type: none">• ブレンダー: 0.6.0 → 0.6.1 (リリースノート)• deadline-cloud: 0.54.0 → 0.54.1 (リリースノート)

日付	タイトル	説明
2026-02-24	Deadline Cloud ドキュメントに、サポートされているソフトウェアユーザーガイドが含まれるようになりました	Deadline Cloud ユーザーガイドに、サポートされている各アプリケーション専用のサブページが追加され、バージョンの互換性と機能のサポートに関する詳細が提供されます。
2026-02-24	ユーザー別の Deadline Cloud Monitor 使用状況エクスポージャーグループの使用状況	使用状況エクスポージャーを使用して、ユーザーあたりの使用状況パターンとチーム全体の属性コストを分析します。
2026-02-24	タスクチャンキングによりレンドリング効率が向上	Deadline Cloud でタスクチャンキングがサポートされるようになりました。これにより、複数のフレームが 1 回のタスク実行にグループ化されます。この機能は、フレームごとに 1 回ではなく、チャンクごとに 1 回アプリケーションとシーンをロードすることで、オーバーヘッドを削減します。デフォルトのチャンクサイズを指定するか、Deadline Cloud にターゲットランタイムに基づいてチャンクサイズを動的に調整させることができます。
2026-02-19	送信者インストーラ v2026-02-19 更新 Maya 送信者	Autodesk Maya 送信者を 0.15.12 から 0.15.13 に更新する新しい送信者インストーラがリリースされました。

日付	タイトル	説明
2026-02-13	OpenJD 仕様で、RFC レビュー用に Claude と Kiro スキルを追加	Open Job Description 仕様リポジトリに、RFC 提案の AI 支援によるレビューのための Kiro スキルが追加され、完全性、明確性、教義の整合性、既存の仕様との互換性が確認されました。
2026-02-13	Deadline Cloud for 3ds Max が AI 支援開発に Kiro パワーを導入	Deadline Cloud for 3ds Max リポジトリに、AI 支援のセットアップ、設計、開発ワークフローを提供する Kiro パワーと、組み込みのガードレールとベストプラクティスが含まれるようになりました。
2026-02-06	ジョブがアクセスコントロールのタグ付けをサポート	ジョブリソースで、タグ付けと属性ベースのアクセスコントロール (ABAC) がサポートされるようになりました。IAM ポリシーは、条件キーを使用してジョブタグを参照し、タグベースの認可パターンを有効にすることができます。たとえば、GetJob API コールを特定のチームタグを持つジョブに制限します。

日付	タイトル	説明
2026-02-05	IAM Identity Center によるマルチリージョンレプリケーションのサポート	Deadline Cloud は、IAM Identity Center のマルチリージョンレプリケーション機能をサポートするようになりました。これにより、スタジオは Identity Center インスタンスと比較して Deadline Cloud をセットアップする場所をより柔軟に設定できます。Studio は、レンダリングのニーズに合ったリージョンに Deadline Cloud フォームを作成できますが、管理者はプライマリリージョンから Identity Center を管理し続けます。
2026-02-04	新しいサンプルジョブバンドルが FLUX.2" LoRA トレーニングをデモンストレーション	20 ~ 50 個のイメージを使用して FLUX.2" モデルでカスタム LoRA アダプターをトレーニングする方法を示すサンプルジョブバンドルが利用可能になりました。これにより、深層機械学習の専門知識を必要とせずに、製品、キャラクター、ブランドアセットのパーソナライズされたイメージジェネレーターを作成できます。LoRA ファインチューニングアプローチは、トレーニングが効率的でチーム間で簡単に共有できる、小型でポータブルなモデルアダプターを作成します。

日付	タイトル	説明
2026-01-29	新しい V-Ray Standalone タイルレンダリングジョブバンドルが利用可能に	エクスポートされた V-Ray シーンのタイルレンダリング用の新しいジョブバンドルが利用可能になりました。このジョブバンドルは、レンダリングファーム全体で並行して処理できるタイルに分割することで、高解像度イメージの効率的なレンダリングを可能にします。3ds Max と V-Ray を使用して V-Ray シーンをローカルにエクスポートし、Windows を使用する代わりに、このバンドルを使用して Linux ワーカーに送信できるようになりました。
2026-01-27	送信後にジョブ名と説明を編集する	Deadline Cloud は、送信後のジョブ名と説明の編集をサポートできるようになりました。この新しい機能により、名前を更新したり、説明フィールドに便利な追跡の詳細を追加したりすることで、送信後のジョブの整理と識別が容易になります。
2026-01-22	サービスマネージドフリートでの Maya の Redshift 2026 サポート	Redshift 2026 は、Deadline Cloud for Maya を使用する Linux サービスマネージドフリートでサポートされるようになりました。

日付	タイトル	説明
2026-01-22	Foundry Nuke CopyCat を使用して機械学習モデルをトレーニングする	Deadline Cloud が Foundry Nuke CopyCat と統合され、クラウドでビジュアルエフェクトの ML トレーニングジョブを実行できるようになりました。CopyCat はサンプルフレームから調整を学習し、シーケンス全体に適用します。Deadline Cloud レンダーファームにトレーニングジョブを送信し、ワークロードを並行してスケールリングして、アーティストワークステーションを解放します。
2026-01-15	SDKsジョブ完了のウェーターを導入	Deadline Cloud SDKs に、ジョブステータスのポーリングを簡素化する JobComplete ウェーターと JobSucceeded ウェーターが含まれるようになりました。JobComplete ウェーターは、ジョブが終了状態 (SUCCEEDED、FAILED、または CANCELED) に達するまでポーリングし、JobSucceeded ウェーターはジョブが成功するまでポーリングします。これらのウェーターにより、カスタムポーリングロジックを記述する必要がなくなるため、ジョブの完了に依存する自動化ワークフローを簡単に構築できます。

日付	タイトル	説明
2026-01-15	<u>予算がアクセスコントロールのタグ付けをサポート</u>	<p>Deadline Cloud のお客様は Budget リソースにタグを適用し、属性ベースのアクセスコントロール (ABAC) を使用してきめ細かなアクセス許可を管理できるようになりました。この新しい機能により、お客様はタグを使用して Deadline Cloud 予算へのアクセスを整理、管理、制御できるため、AWS リソース全体で一貫した認可パターンが可能になります。お客様は作成時に予算にタグを付け、IAM ポリシーでこれらのタグを使用して、タグ値に基づいて特定の予算にアクセスできるユーザーを制御できるようになりました。</p>

日付	タイトル	説明
2026-01-15	Deadline Cloud Monitor 検索で複数選択フィルタリングを追加	Deadline Cloud モニターを使用する場合、ユーザー名やジョブステータスなど、検索フィルターに最大 16 個の値を選択できるようになりました。これにより、複数のユーザー間でジョブをすばやく検索したり、複数のステータスを一度にフィルタリングしたりできます。この機能は、ジョブ、ステップ、タスク、ワーカーの新しい StringListFilterExpression を通じて Deadline Cloud API でも使用できます。
2026-01-07	Direct Deadline Cloud Monitor と送信者インストーラのダウンロードリンクが利用可能	ユーザーは Deadline Cloud Monitor デスクトップアプリケーションと送信者インストーラを Deadline Cloud ドキュメントから直接ダウンロードできるようになりました。これにより、AWS コンソールにアクセスできないユーザーは、Deadline Cloud の使用を開始するために必要なソフトウェアをダウンロードできるようになります。

日付	タイトル	説明
2025-12-19	送信者インストーラ v2025-12-19 がリリースされました	<p>次のコンポーネントを更新する新しい送信者インストーラがリリースされました。</p> <ul style="list-style-type: none">• cinema-4d: 0.9.0 → 0.9.2 (リリースノート)• deadline-cloud: 0.53.3 → 0.54.0 (リリースノート)• nuke: 0.18.13 → 0.18.14 (リリースノート)
2025-12-17	Deadline Cloud Monitor 1.1.7 が統合ジョブ送信を導入	<p>最新の Deadline Cloud Monitor デスクトップアプリケーションリリースには以下が含まれます。</p> <ul style="list-style-type: none">• Deadline Cloud Monitor デスクトップアプリケーションから直接ジョブを送信するためのサポート。• ワークステーションのセットアップが簡単になりました。• プロキシのサポートが改善されました。• Deadline Cloud プロファイル設定ファイルとの間で読み書きする際のエッジケースのバグ修正。

日付	タイトル	説明
2025-12-11	デベロッパーガイドに AI エージェントガイダンスを追加	Deadline Cloud 開発者ガイドには、AI エージェントを使用してジョブバンドルを記述し、conda パッケージを開発し、ジョブをより効率的にトラブルシューティングするためのベストプラクティスが含まれています。
2025-12-10	Autodesk VRED 送信者ユーザーガイドのリリース	Autodesk VRED の Deadline Cloud 送信者のドキュメントが利用可能になりました。このガイドでは、送信者をインストールし、レンダリングジョブを Deadline Cloud に送信する方法について説明します。これにより、VRED ユーザーはクラウドレンダリングをすばやく開始できます。
2025-12-10	デベロッパーガイドに LicensesInUse メトリクスセクションを追加	Deadline Cloud デベロッパーガイドに LicensesInUse メトリクスに関する情報が追加されました。このメトリクスは、ジョブがフリート全体で現在使用しているライセンスの数をモニタリングするのに役立ちます。この情報を使用して、ライセンスの使用を最適化し、ワークロードをスケールアウトするときにライセンスが不足しないようにできます。

日付	タイトル	説明
2025-12-10	サービスマネージドフリートに Cinema 4D 2026.1 サポートを追加	<p>Maxon Cinema 4D 2026.1 が Linux および Windows のサービスマネージドフリートでサポートされるようになりました。このリリースには、Redshift 2026.2.0 が含まれています。Cinema 4D のすべてのバージョンでクロスプラットフォームフォントレンダリングのサポートも追加されました。このリリースでは、最新の Cinema 4D 機能を使用できます。また、起動時間を短縮し、Linux ワーカーのコストを削減しながら Windows からジョブを送信する場合など、クロスプラットフォーム設定でカスタムフォントを使用することもできます。</p>
2025-12-09	Autodesk Maya 送信者がセットアップと使用に関するドキュメントを拡張	<p>Autodesk Maya の Deadline Cloud Submitter 用の新しいセットアップと使用に関するドキュメントが追加されました。</p>

日付	タイトル	説明
2025-12-09	After Effects 送信者 0.4.4 が macOS のインストールとフォントのサポートを改善	<p>After Effects 送信者が macOS のユーザー設定ディレクトリに自動的にインストールされるようになり、手動インストールが不要になりました。このリリースでは、ほとんどの TrueType Collection (TTC) フォントファイルのサポートも追加され、これらのフォントを使用するジョブを送信およびレンダリングできます。これらの改善により、After Effects ユーザーのフォント互換性の設定と拡張が簡素化されます。</p>
2025-12-08	ユーザーガイドのリリースノートページの起動	<p>Deadline Cloud の機能、アプリケーション、統合、サンプル、ドキュメントの主な変更はすべて、今後のユーザーガイドのリリースノートページに表示されるようになりました。以前の主要な Deadline Cloud リリースは、AWS What's New および CLI/Worker/integration 固有のリリースノートの Deadline Cloud Github 組織 のリポジトリにあります。</p>

AWS 用語集

最新の AWS 用語については、「AWS の用語集 リファレンス」の [AWS 「用語集」](#) を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。