ユーザーガイド

AWS Data Exports

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Data Exports: ユーザーガイド

Copyright © 2025 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

とは AWS Data Exports 1
CUR からデータエクスポート CUR 2.0 への移行 3
CUR スキーマを使用したエクスポートの作成 5
CUR 2.0 のエクスポートを新しいスキーマで作成する 7
データエクスポートの作成 9
データエクスポート用の Amazon S3 バケットのセットアップ 10
標準エクスポートの作成 12
コストと使用状況ダッシュボードの作成 16
レガシー CUR エクスポートの作成 17
データクエリ - SQL クエリとテーブル設定 20
SQL クエリ 21
テーブル設定 23
データエクスポートの表示と管理 24
エクスポート配信について 25
エクスポート S3 の親ディレクトリ構造 25
エクスポートの更新 27
エクスポートの上書きと新規作成 27
エクスポートデータのファイル名とチャンク 28
概要 28
エクスポートの詳細の編集 30
エクスポートタグの編集 30
エクスポートの削除 31
AWS Organizations でのデータエクスポートの使用 31
データエクスポートテーブルディクショナリ 33
コストと使用状況レポート (CUR) 2.0 33
テーブル設定 34
AWS Organizations のサポート 36
CUR 2.0 の列グループ 37
請求書列 37
コストカテゴリ列 39
割引列 40
アイデンティティ列 42
明細項目列 43
料金列 55

製品列 57
予約列 62
リソースタグ列 70
Savings Plan 列 70
分割明細項目列 73
コスト最適化の推奨事項 79
テーブル設定 79
サービスリンクロール 81
AWS Organizations のサポート 82
コスト最適化の推奨事項列 82
AWS 列を含む FOCUS 1.0 85
テーブル設定 85
AWS Organizations のサポート 85
AWS 列を含む FOCUS 1.0 86
列の適合ギャップを含む AWS FOCUS 1.0 94
AWS 列を含む FOCUS 1.0 (プレビュー) 96
プレビューメモ 97
テーブル設定 97
AWS Organizations のサポート 97
AWS 列を含む FOCUS 1.0 (プレビュー) の列 98
AWS 列を含む FOCUS 1.0 (プレビュー) の適合ギャップ 106
コストと使用状況ダッシュボード 111
テーブル設定 112
AWS Organizations のサポート 112
コストと使用状況ダッシュボード 112
炭素排出量 126
履歴データ 126
テーブル設定 126
アクセス許可 126
モデルバージョン 126
AWS Organizations のサポート 126
炭素排出量列 127
データエクスポートの処理 131
Amazon Athena の設定 131
Amazon Redshift の設定 132
CUR 2.0 の処理するための推奨 SQL クエリ 132

炭素排出量データのエクスポートを処理するための推奨 SQL クエリ 132
コストと使用状況ダッシュボードについて 134
コストと使用状況レポート (CUR) について 136
炭素排出量データのエクスポートについて 137
セキュリティおよびアクセス許可 138
データエクスポートの Identity and Access Management 138
ポリシーの例 142
データエクスポートにおけるデータ保護 143
S3 セキュリティのベストプラクティス 144
S3 でのデータ暗号化 144
クォータと制限 145
トラブルシューティング 146
一般的なトラブルシューティング 146
エクスポートが異常なのはなぜですか。 146
SQL ステートメントがデータエクスポートで受け入れられないのはなぜですか。 148
データエクスポート内で Athena を設定するための事前定義された SQL スクリプトが見つかりません。 148
エクスポートパーティションの 1 つが空なのはなぜですか。 148
Amazon S3 バケットにレポートファイルがないのはなぜですか？ 148
CUR 2.0 のトラブルシューティング 148
CUR で使用可能な一部の列が CUR 2.0 に表示されないのはなぜですか？ 149
従来のコストと使用状況レポートはどうなりますか。廃止されますか？ 149
CUR 2.0 のエクスポートを作成すると、レガシー CUR に影響がありますか？ 150
データエクスポートと CUR テーブルを使用する IAM アクセス許可があっても、CUR 2.0 のエクスポートを作成できないのはなぜですか？ 150
レガシー CUR 列と同じ CSV 形式でデータエクスポートを作成しようとすると、「無効な QueryStatement」エラーが発生します。このエラーを解消するにはどうすればよいですか? 150
データエクスポート CUR 2.0 に移行した後、レガシー CUR エクスポートと CUR 2.0 エクスポートを同時に取得できますか? 150
CUR 2.0 のエクスポートを作成しようとすると、「このアカウントはこのテーブルに対してエクスポートを作成できません」というエラーが表示されます。CUR 2.0 エクスポートを作成できないのはなぜですか。 150
コストと使用状況ダッシュボードのトラブルシューティング 151
コストと使用状況ダッシュボードを作成した直後にエクスポートに失敗したのはなぜですか。 151

ダッシュボードにアクセスできないのはなぜですか。 151
ダッシュボードを表示しようとすると、QuickSight アカウントの登録を解除するためにコンソールの管理ページに移動するのはなぜですか。 152
作成したばかりのコストと使用状況ダッシュボードにデータが表示されないのはなぜですか? 152
コストと使用状況ダッシュボードに履歴データが表示されないのはなぜですか? 152
QuickSight ダッシュボードのリンクがデータエクスポートコンソールページから消えたのはなぜですか。 153
CUR 2.0 でリソースタグを視覚化するように Amazon QuickSight を設定するにはどうすればよいですか？ 153
コストと使用状況レポートのトラブルシューティング 153
コストと使用状況レポートのデータが他の請求情報とコスト管理機能に表示されるデータと一致しないのはなぜですか？ 154
レポート設定を変更した後にデータをバックフィルするにはどうすればよいですか？ 155
Amazon S3 のレポートファイルフォルダが名前のないフォルダに保存されるのはなぜですか？ 155
レポートにリソース IDsを含めるオプションを選択できないのはなぜですか？ 155
Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で機能しないのはなぜですか? または Amazon Athena の Amazon Redshift クエリが機能しないのはなぜですか？ 156
レポートに含まれる列が前月から変更された理由 156
レポートの列が変更された後、クエリまたはテーブルが機能しないのはなぜですか？ 157
レポートにクエリを実行するにはどうすればよいですか？ 157
Amazon EC2 Dedicated Host の請求データはどこにありますか？ 157
Amazon EC2 Elastic IP アドレスの請求データを解釈するにはどうすればよいですか？ 158
非ブレンドレートとブレンドレートまたはコストは、一括請求でどのように異なりますか？ 158
レポートの一部の明細項目でブレンドレートまたはブレンドコストが 0 であるのはなぜですか？ 158
レポートですべての前払いリザーブドインスタンスはどのように償却されますか？ 159
炭素排出量データエクスポートのトラブルシューティング 159
データエクスポートと CUR 2.0 テーブルを使用する IAM アクセス許可があっても、炭素排出量テーブルのエクスポートを作成できないのはなぜですか？ 159
組織内の一部のメンバーアカウントの炭素排出量データが表示されないのはなぜですか？ 160
S3 バケット内のいずれかのファイルが空であるのはなぜですか？ 160

使用状況データがある場合、一部のリージョンとサービスで S3 エクスポートに炭素排出量ゼロが表示されるのはなぜですか？ ... 160
過去のデータバックフィルは、炭素排出量のデータエクスポートで利用できますか？ ... 160
S3 バケットに履歴データが表示されないのはなぜですか？ ... 160
レポートの設定を変更しました。データを新しい設定でバックフィルできますか？ ... 161
従来のコストと使用状況レポート ... 162
AWS コストと使用状況レポートとは ... 162
コストと使用状況レポートの仕組み ... 162
レポートタイムライン ... 163
レポートファイル ... 163
レポート列 ... 164
レポートの使用 ... 164
コストと使用状況レポートを作成する ... 164
コストと使用状況レポート用に Amazon S3 バケットをセットアップする ... 165
レポートの作成 ... 167
レポートの表示と管理 ... 169
最新のレポートバージョンを表示する ... 170
確定レポートの表示 ... 171
レポートのバージョンを確認する ... 172
レポートの編集 ... 176
のコストと使用状況レポートの使用 AWS Organizations ... 178
Athena を使用したレポートのクエリ ... 180
CloudFormation を使用した Athena のセットアップ ... 180
Athena の手動セットアップ ... 183
Athena クエリの実行 ... 186
その他のリソース ... 187
Billing Conductor AWS を使用した CUR の設定 ... 190
Billing Conductor CUR AWS と標準 CUR AWS の違い ... 190
請求グループのプロフォーマコストと使用状況レポートを作成します ... 190
データディクショナリ ... 192
ID の詳細 ... 193
請求の詳細 ... 193
明細項目の詳細 ... 195
予約の詳細 ... 202
料金詳細 ... 212
製品の詳細 ... 214

リソースタグの詳細 ... 247
Savings Plans の詳細 ... 247
Cost Categoriesの 詳細 ... 253
ディスカウントの詳細 ... 254
分割明細項目の詳細 ... 254
ユースケース ... 259
Savings Plans について ... 259
予約を理解する ... 263
データ転送料金について ... 276
分割コスト配分データを理解する ... 279
レガシー請求レポートについて ... 297
請求明細レポート ... 298
DBR から AWS CUR への移行 ... 299
未使用予約のコストの確認 ... 304
毎月のレポート ... 314
毎月のコスト配分レポート ... 314
AWS 使用状況レポート ... 315
トラブルシューティング ... 316
Amazon S3 バケットにはレポートファイルはありません ... 317
レポートデータパーティションの 1 つが空です ... 317
コストと使用状況レポートのデータが、他の請求とコスト管理機能のデータと一致しない ... 317
レポートの設定を変更したので、データをバックフィルしたいです。 ... 318
Amazon S3 のレポートファイルフォルダが名前のないフォルダにあります ... 318
レポートにリソース ID を含めるオプションを選択できない ... 318
Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で動作しない、またはAmazon Redshift クエリが Amazon Athena で動作しない ... 318
レポートに含まれる列が前月と変わっています。 ... 319
レポート内の列が変更されたため、レポートに基づくクエリやテーブルが機能しない ... 320
レポートをクエリする方法を知りたい ... 320
Amazon EC2 専用ホストの請求データが見つかりません ... 320
Amazon EC2 Elastic IP アドレスの請求データがわかりません ... 320
一括請求を使っているが、非ブレンドとブレンドのレートまたはコストの違いがわからない ... 320
レポートの一部の明細項目には、ブレンドレートまたはブレンドコストが 0 のものがあります。 ... 321

私のレポートでは、全額前払いリザーブドインスタンスがどのように償却されているのかわかりません。 321
セキュリティおよびアクセス許可 322
クォータと制限 322
コストと使用状況レポート 322
ヘルプの利用 323
ドキュメント履歴 325
..... cccxxvii

とは AWS Data Exports

AWS Data Exports では、基本的な SQL を使用して請求およびコスト管理データのエクスポートと炭素排出量データのエクスポートを作成し、Amazon QuickSight と統合してデータを視覚化できます。

エクスポートは、 AWS Billing and Cost Management コンソール、 AWS CLI、または AWS SDK を使用して作成できます。コンソールでは、カスタム列選択を使用できます。CLI または AWS SDK では、SQL AWS クエリの書き込み、列の選択、行のフィルタリング、列の名前変更を行うことができます。これにより、処理したいデータのみを選択したり、機密性の高いコスト情報を削除したり、エクスポートの出力スキーマを制御したりできます。

エクスポートには 3 つのタイプがあります。

- 標準データエクスポート。4 つの異なるテーブルから選択できます。
  - コストと使用状況レポート (CUR) 2.0

    > **Note**
    > コストと使用状況レポート 2.0 (CUR 2.0) は、詳細な AWS コストと使用状況データを受け取るための新しく推奨される方法です。CUR 2.0 には、以前のコストと使用状況レポート (CUR) からいくつかの改善点があります。詳細については、「Migrating from CUR to CUR 2.0 in Data Exports」を参照してください。

  - コスト最適化の推奨事項 (Cost Optimization Hub から)
  - AWS 列を含む FOCUS 1.0
  - 炭素排出量
- [コストと使用状況ダッシュボード]: Amazon QuickSight へのエクスポートと統合で、事前に構築されたコストと使用状況のダッシュボードを展開します。
- [レガシーデータのエクスポート]: レガシーコストと使用状況レポート (CUR) のエクスポート。ただし、レガシー CUR には、データエクスポートアクション (「データエクスポートアクションのリファレンス」を参照) とは異なる一連のアクション (「CUR アクションリファレンス」を参照) を使用してアクセスします。

データエクスポートには次の利点があります。

- 利用可能な最も詳細なデータを使用して、定期的なエクスポートを作成し、Amazon S3 に保存します。
- 列選択と行フィルターを使用して SQL クエリでデータエクスポートをカスタマイズします。
- 必要な列だけを含む一貫性のあるスキーマでエクスポートを作成できます。
- 特定のリンクされた AWS アカウント IDs に関連する機密データや料金を削除します。
- 必要な列または行のみを選択することで、エクスポートのサイズを縮小します。
- コストデータとカーボンフットプリントデータのエクスポートを自動化して、ダウンストリーム分析をサポートします。

データエクスポートを開始するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。

[データエクスポート] ページから、新しいエクスポートの作成や、既存のエクスポートの管理、Amazon QuickSight と統合して事前に構築されたコストと使用状況ダッシュボードをデプロイするエクスポートを作成したりできます。

データエクスポートページから Customer Carbon Footprint Tool とAWS 使用状況レポートにアクセスすることもできます。

CUR からデータエクスポート CUR 2.0 への移行

AWS Data Exports では、コストと使用状況レポート 2.0 (CUR 2.0) のエクスポートを作成できます。CUR 2.0 テーブルは、コストと使用状況レポート (CUR) と同じ情報を提供し、いくつかの改善点を含んでいます。データエクスポートにより、CUR の処理に使用していたデータパイプラインと下位互換性のある CUR 2.0 エクスポートを作成できます。

CUR 2.0 は CUR と比べて以下の点が改善されています。

- 一貫性のあるスキーマ: CUR 2.0 には固定の列セットが含まれていますが、CUR に含まれる列は、 AWS のサービスの使用状況、コストカテゴリ、およびリソースタグに応じて毎月異なる場合があります。
- ネストデータ: CUR 2.0 は、CUR の特定の列を、折りたたんだ列のキーと値のペアを含む個別の列に折りたたむことで、データスパース性を低減します。オプションで、データエクスポート内のネストされたキーを、元の CUR スキーマおよびデータと一致する個別の列としてクエリできます。
- 追加列: CUR 2.0 には、bill_payer_account_name と line_item_usage_account_name という 2 つの列が追加されています。

次の表は、CUR 2.0 とレガシー CUR の違いをより詳細にまとめたものです。

	CUR 2.0	レガシー CUR
データスキーマ	スキーマを修正しました。 完全な列リストについては、「コストと使用状況レポート (CUR) 2.0」を参照してください。	AWS 使用状況とアクティビティに基づく動的スキーマ。 部分列リストについては、「データディクショナリ」を参照してください。

限定列	`bill_payer_account_name`
	
	`line_item_usage_account_name`
		なし

	CUR 2.0	レガシー CUR
エクスポートのカスタマイズ	列の選択、行フィルタリング、列のエイリアシング (名前変更) の基本的な SQL を有効にします。 サポートされている SQL 構文の詳細については、「データクエリ」を参照してください。	サポート外。必要なビューを作成するには、Athena/QuickSight を手動で設定する必要があります。

キーと値のペアを持つネストされた列	`resource_tags`
	
	`cost_category`
	
	`product`
	
	`discount`
		ネストされた列はありません。
		
		CUR 2.0 の 4 つのネストされた列は、レガシー CUR の別々の列に分割されます (例: `resource_tags_user_creator`)。

ファイル配信先	S3 バケット	S3 バケット
出力ファイル形式	GZIP、Parquet	ZIP、GZIP、Parquet
他の AWS サービスとの統合	Amazon QuickSight	Amazon Athena、Amazon Redshift、Amazon QuickSight
Amazon CloudFormation のサポート	はい 詳細については、「AWS CloudFormation ユーザーガイド」の「AWS データエクスポートのリソースタイプのリファレンス」を参照してください。	はい 詳細については、「AWS CloudFormation ユーザーガイド」の「AWS コストと使用状況レポートのリソースタイプのリファレンス」を参照してください。

	CUR 2.0	レガシー CUR
タグとコストカテゴリデータ	タグとコストカテゴリ名は正規化され、特殊文字とスペースが削除されます。正規化後にタグまたはコストカテゴリが競合する場合は、1 つの値のみが保持されます。詳細については、「列名」を参照してください。	動作は、レガシー CUR Parquet と CSV ファイル形式によって異なります。 レガシー CUR Parquet: タグとコストカテゴリ名は正規化され、特殊文字とスペースが削除されます。正規化後にタグまたはコストカテゴリが競合する場合は、1 つの値のみが保持されます。詳細については、「列名」を参照してください。 レガシー CUR CSV: タグとコストカテゴリ名は変更されません。

CUR 2.0 のスキーマの詳細については、「Data Exports table dictionary」を参照してください。

データエクスポートで CUR 2.0 に移行するには、次の 2 つの方法があります。

- 方法 1: CUR スキーマを使用して SQL クエリでエクスポートを作成する
- 方法 2: CUR 2.0 のエクスポートを新しいスキーマで作成する

方法 1: CUR スキーマを使用して SQL クエリでエクスポートを作成する

SQL クエリを使用してエクスポートを作成できます。エクスポートスキーマは、現在 CUR で受け取っているものと一致します。これは、 AWS API または SDK を使用して行います。

1. 現在の CUR に合わせるために必要な (a) 列のリストと (b) CUR コンテンツ設定 ([リソース IDを含める]、[コスト配分データの分割]、および [時間粒度]) を決定します。

1. 列のリストは、いずれかのCUR ファイルのスキーマを表示するか、マニフェストファイルに移動してそこから列のリストを抽出することで決定できます。
2. CUR コンテンツの設定は、コンソールの [データエクスポート] に移動し、CUR エクスポートを選択して詳細を表示することで決定できます。

2. `COST_AND_USAGE_REPORT` という名前の CUR 2.0 テーブルから特定した列を選択する SQL クエリを書き込みます。

   1. CUR 2.0 テーブルの列名はすべてスネークケースです (例: `line_item_usage_amount`)。SQL ステートメントでは、以前の列名をスネークケースに変換することが必要な場合があります。
   2. SQL ステートメントでは、CUR 2.0 でネストされた列を選択するために、すべての `resource_tag` と `cost_category` 列、そして特定の `product` と `discount` 列をドット演算子に変換する必要があります。例えば、CUR 2.0 で `product_from_location` 列を選択するために、`product.from_location` を選択する SQL ステートメントを書き込みます。

      例: `SELECT product.from_location FROM COST_AND_USAGE_REPORT`

      これにより、`product` マップ列の `from_location` 列が選択されます。
   3. デフォルトでは、ドット演算子で選択された列には、属性によって名前が付けられます (例: `from_location`)。既存の CUR と一致させるには、以前と同じになるように列のエイリアスを宣言する必要があります。

      例: `SELECT product.from_location AS product_from_location FROM COST_AND_USAGE_REPORT`

      ネストされた列について詳しくは、「Data Exports table dictionary」を参照してください。

3. ステップ 1 で特定した CUR コンテンツ設定を `CreateExport` API のテーブル設定形式に書き込みます。次のステップでは、これらのテーブル設定をデータクエリで提供する必要があります。
4. データエクスポート用の AWS SDK/CLI で、 `CreateExport` API を使用して SQL クエリとテーブル設定をデータクエリフィールドに入力します。

   1. ターゲットの Amazon S3 バケットや上書き設定など、配信設定を指定します。以前と同じ配送設定を選択することをお勧めします。必須フィールドの詳細については、「AWS Billing and Cost Management API リファレンス」の「AWS Data Exports」を参照してください。

2. データエクスポートがバケットに書き込めるように、ターゲット Amazon S3 バケットのアクセス許可を更新します。詳細については、「Setting up an Amazon S3 bucket for data exports」を参照してください。

5. CUR 2.0 が配信されている Amazon S3 バケット内のディレクトリからデータを読み取るようにデータインジェストパイプラインに指示します。

方法 2: CUR 2.0 のエクスポートを新しいスキーマで作成する

CUR 2.0 のエクスポートは、ネストされた列と追加の列からなる新しいスキーマを使用して作成できます。ただし、これらの新しい列を処理するには、現在のデータパイプラインを調整する必要があります。これを行うには、 コンソール、 AWS API、または SDK を使用します。

1. 現在の CUR に合わせるために必要な CUR コンテンツ設定 ([リソース ID を含める]、[コスト配分データの分割]、[時間粒度]) を決定します。

   - CUR コンテンツの設定は、コンソールの [データエクスポート] に移動し、CUR エクスポートを選択して詳細を表示することで決定できます。

2. データエクスポートコンソールページ (オプション A) または AWS SDK/CLI (オプション B) のいずれかを使用して、「コストと使用状況レポート」テーブルからすべての列を選択する CUR 2.0 のエクスポートを作成します。

3. (オプション A) コンソールでエクスポートを作成するには:

   1. ナビゲーションペインで [データエクスポート] を選択します。
   2. [データエクスポート] ページで、[作成] を選択します。
   3. [標準データエクスポート] を選択します。

      コストと使用状況レポート (CUR 2.0) テーブルでは、すべての列がデフォルトで選択されています。

   4. ステップ 1 で特定した CUR コンテンツ設定を指定します。
   5. [データテーブル配信オプション] で、オプションを選択します。
   6. [作成] を選択します。

4. (オプション B) AWS API/SDK を使用してエクスポートを作成するには、まず `COST_AND_USAGE_REPORT` テーブルのすべての列を選択するクエリを書き込みます。

   1. `GetTable` API を使用して列の全リストを決定し、スキーマ全体を受け取ります。

2. ステップ 1 で特定した CUR コンテンツ設定を `CreateExport` API のテーブル設定形式に書き込みます。
3. `CreateExport` API を使用して SQL クエリとテーブル設定を `data-query` フィールドに入力します。
4. ターゲットの Amazon S3 バケットや上書き設定など、配信設定を指定します。以前と同じ配送設定を選択することをお勧めします。必須フィールドの詳細については、「AWS Billing and Cost Management API リファレンス」の「AWS Data Exports」を参照してください。
5. データエクスポートがバケットに書き込めるように、ターゲット Amazon S3 バケットのアクセス許可を更新します。詳細については、「Setting up an Amazon S3 bucket for data exports」を参照してください。

5. CUR 2.0 が配信されている Amazon S3 バケット内のディレクトリからデータを読み取るようにデータインジェストパイプラインに指示します。

   また、データインジェストパイプラインとビジネスインテリジェンスツールを更新して、キー値がネストされた以下の新しい列を処理する必要があります (`product`、`resource_tags`、`cost_category`、および `discounts`)。

データエクスポートの作成

Billing and Cost Management コンソールのデータエクスポートページを使用して、標準エクスポート、コストと使用状況ダッシュボードエクスポート、レガシーエクスポートの 3 つの異なるタイプのデータエクスポートを作成できます。

テーブルごとに作成できるエクスポートの数には、次の制限があります。

- コストと使用状況レポート 2.0 (CUR 2.0): エクスポート 5 件
- コスト最適化の推奨事項: エクスポート 2 件
- 列を含む AWS FOCUS 1.0: 2 エクスポート
- コストと使用状況ダッシュボード: エクスポート 2 件
- 炭素排出量: 2 エクスポート

詳細については、「クォータと制限」を参照してください。

コンソールでエクスポートを作成し、エクスポートするテーブルを選択するか、 AWS SDK/CLI でエクスポートを作成し、必要なデータテーブルから列選択と行フィルターの SQL クエリを定義して、数分でエクスポートを設定します。

コンソールでエクスポートを作成する場合、データエクスポートストレージ用の Amazon S3 バケットを作成できます。 AWS SDK/CLI でエクスポートを作成する場合は、事前に正しいバケットポリシーを使用して Amazon S3 バケットを作成する必要があります。詳細については、「データエクスポート用の Amazon S3 バケットのセットアップ」を参照してください。

新しいデータエクスポートを作成すると、データエクスポートは Amazon S3 バケットへのデータのエクスポートを開始します。

> **Note**
>
> が Amazon S3 バケットへのエクスポートの配信を開始する AWS までに最大 24 時間かかる場合があります。 Amazon S3 配信が開始されたら、S3 バケットの請求とコスト管理のエクスポート出力を少なくとも 1 日に 1 回、炭素排出量のエクスポート出力を少なくとも 1 か月に 1 回 AWS 更新します。実際の更新レートは、さまざまな要因によって異なる場合があります。

トピック

- データエクスポート用の Amazon S3 バケットのセットアップ
- 標準エクスポートの作成
- コストと使用状況ダッシュボードの作成
- レガシー CUR エクスポートの作成
- データクエリ - SQL クエリとテーブル設定

データエクスポート用の Amazon S3 バケットのセットアップ

データエクスポートを受信して保存するには、 AWS アカウントに Amazon S3 バケットが必要です。コンソールでエクスポートを作成するときに、所有している既存の S3 バケットを選択することも、新しいバケットを作成することもできます。いずれの場合でも、以下のデフォルトの S3 バケットポリシーの適用を確認して確定する必要があります。Amazon S3 コンソールでこのポリシーを編集したり、エクスポートを作成した後で S3 バケットの所有者を変更したりすると、データエクスポートがエクスポートを配信できなくなります。S3 バケットに保存されているエクスポートデータは、標準の Amazon S3 レートで課金されます。詳細については、「クォータと制限」を参照してください。

> **Note**
>
> エクスポートを作成するアカウントは、エクスポート AWS を送信する S3 バケットも所有している必要があります。別のアカウントが所有する S3 バケットに配信するようにエクスポートを設定することはできません。

データエクスポートの作成時に、次のポリシーがすべての S3 バケットに適用されます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EnableAWSDataExportsToWriteToS3AndCheckPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "billingreports.amazonaws.com",
                    "bcm-data-exports.amazonaws.com"
```

```
                ]
            },
            "Action": [
                "s3:PutObject",
                "s3:GetBucketPolicy"
            ],
            "Resource": [
                "arn:aws:s3:::${bucket_name}/*",
                "arn:aws:s3:::${bucket_name}"
            ],
            "Condition": {
                "StringLike": {
                    "aws:SourceAccount": "${accountId}",
                    "aws:SourceArn": [
                        "arn:aws:cur:us-east-1:${accountId}:definition/*",
                        "arn:aws:bcm-data-exports:us-east-1:${accountId}:export/*"
                    ]
                }
            }
        }
    ]
}
```

この S3 バケットポリシーにより、データエクスポートがエクスポートを作成したアカウントに代わって S3 バケットにのみエクスポートを配信できます。また、データエクスポートでは、エクスポートを作成したアカウントが S3 バケットをまだ所有していることを確認できます。

- エクスポートを S3 バケットに配信するには、その S3 バケットの書き込みアクセス許可 AWS が必要です。これを行うために、S3 バケットポリシーは、データエクスポートサービス (`bcm-data-exports.amazonaws.com`) に、ユーザーが所有する S3 バケット (`arn:aws:s3:::<EXAMPLE-BUCKET>/*`) にレポートを配信 (`s3:PutObject`) するためのアクセス許可を付与します。
- データエクスポートが S3 バケットへの書き込みをリクエストするたびに、エクスポートを作成したアカウントのアカウント ID を提供する必要があります。`aws:SourceArn` と `aws:SourceAccount` という条件キーによりこれが強制的に適用されます。
- この S3 バケットポリシーは、配信後のコストと使用状況レポートなど、S3 バケット内のオブジェクトを読み取りまたは削除するための AWS アクセス許可を付与しません。

アクセスコントロールリスト (ACL) が有効になっている Amazon S3 バケットでは、データエクスポートは配信時には `BucketOwnerFullControl` ACL をレポートに適用します。デフォルトでは、

これらのレポートなどの Amazon S3 オブジェクトは、それらを作成したユーザーまたはサービスプリンシパルのみが読み取ることができます。ユーザーまたは S3 バケット所有者にレポートを読み取るためのアクセス許可を付与するには、 AWS が `BucketOwnerFullControl` ACL を適用する必要があります。ACL はこれらのレポートに対する `Permission.FullControl` を S3 バケット所有者に付与します。ただし、ACL を無効にし、S3 バケットポリシーを使用してアクセスを制御することをお勧めします。

> **Note**
>
> 新しく作成された S3 バケットでは、ACL はデフォルトで無効になっています。詳細については、「バケットのオブジェクト所有権のコントロールと ACL の無効化」を参照してください。

[データエクスポート] コンソールページに [無効なバケット] エラーが表示される場合は、レポートセットアップ後にポリシーと S3 バケットの所有権が変更されていないことを確認します。

標準エクスポートの作成

他の処理ツール (Amazon Athena など) を使用して分析できる標準データエクスポートを作成できます。

標準データエクスポートを作成するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. [Create Export] (エクスポートを作成) を選択します。
4. [エクスポートを作成] ページの [エクスポートタイプ] で、[標準データエクスポート] を選択します。
5. [エクスポート名] に、エクスポートの名前を入力します。

   エクスポート名は、最大 128 文字で、一意である必要があります。有効な文字は、a-z、A-Z、0-9、 - (ハイフン)、および _ (アンダースコア) です。
6. [データテーブルの設定] で、エクスポートに含めるデータと列を指定できます。まず、エクスポートするテーブルを選択します。

> **Note**
>
> コスト最適化レコメンデーションテーブルをエクスポートするには、サービスにリンクされたロールが必要です。詳細については、「データエクスポートのサービスにリンクされたロール」を参照してください。
> 炭素排出量テーブルをエクスポートするには、Customer Carbon Footprint Tool とデータにアクセス`sustainability:GetCarbonFootprintSummary`するための IAM アクセス許可が必要です。

列と炭素排出量を含む AWS FOCUS 1.0 を除き、エクスポートにデータを追加するテーブル設定は異なります。

1. CUR 2.0 の場合:
   a. [リソース ID を含める] を選択すると、各リソースの ID をエクスポートに含めることができます。

      > **Note**
      >
      > リソース ID を含めると、リソースごとに個別の明細項目が作成されます。これにより、 AWS 使用量に基づいてエクスポートのサイズが大幅に増加する可能性があります。

   b. 共有リソースの詳細なコストと使用状況を含めるには、[コスト配分データの分割] を選択します (Amazon ECS と Amazon EKS)。

      > **Note**
      >
      > コスト配分データの分割を含めると、リソース (つまり ECS タスクと Kubernetes ポッド) ごとに、個別の明細項目が作成されます。これにより、 AWS の使用状況に基づいて、コストと使用状況レポートのサイズが大幅に増加する可能性があります。

   c. [手動割引形式を有効にする] を選択すると、割引が標準の自動形式ではなく手動割引形式でコストと使用状況レポートに表示されるように変換されます。

> **Note**
> このオプションは、割引自動化プログラムを利用している場合にのみ表示されます。

d. [時間詳細度] では、時間単位、日単位、または月単位で選択して、エクスポート内の明細項目をその時間詳細度で集計します。

2. AWS 列を含む FOCUS 1.0 の場合、テーブル設定はありません。
3. 炭素排出量の場合、テーブル設定はありません。
4. コスト最適化の推奨事項の場合:
   a. [すべての推奨事項を含める]を選択し、互換性のない推奨事項の最小節約額の推奨事項を削除します。
   b. 特定のタイプの推奨事項を除外する場合は、互換性のない推奨事項を削除する前に推奨事項フィルターを追加します。

> **Note**
> これらの設定を Cost Optimization Hub コンソールで指定した場合、Cost Optimization Hub で [エクスポートを作成] を選択すると、データエクスポートに引き継がれます。

7. [列の選択] では、エクスポートに含める列を選択します。不明な場合は、表の一番上にある最初のチェックボックスをオンにして、すべての列を選択します。列を多く選択すると、エクスポートのファイルサイズが大きくなる可能性があります。
8. データテーブル配信オプションで、データエクスポートの更新頻度: 。
   - 請求およびコスト管理データのエクスポートでは、使用できる唯一のオプションは日次 - エクスポートは 1 日に 1 回まで更新されます。
   - 炭素排出量データのエクスポートでは、使用可能なオプションは毎月 - エクスポートは月に 1 回更新されます。各更新では、3 か月前の炭素排出量データが提供されます (例えば、4 月の更新には 1 月のデータが含まれます) 。
9. [圧縮タイプおよびファイル形式] では、エクスポートを次のいずれかから選択します。
   - Parquet − Parquet
   - gzip — text/csv

10. [ファイルバージョニング] では、更新ごとにエクスポートが上書きされるかどうかを決定する次のいずれかを選択します。

    - [既存のデータエクスポートファイルを上書き]: エクスポートを更新するたびに、データパーティション内の前回の配信 (請求期間など) が上書きされます。エクスポートを上書きすると、Amazon S3 ストレージコストを節約できます。

      > **Note**
      > 上書きは、コスト最適化の推奨事項のエクスポートではサポートされていません。

    - [新しいデータエクスポートファイルの作成]: 同じパーティション (請求期間など) の配信であっても、エクスポートを更新するたびに別のディレクトリに書き込まれます。新しいエクスポートバージョンを作成すると、コストと使用状況データの経時的な変化を追跡できます。

11. [データエクスポートストレージ設定] の [S3 バケット] 名で [設定] を選択します。
12. [S3 バケットの設定] ダイアログボックスで、次のいずれかを実行します。

    - 既存のバケットを選択します。
    - [バケットの作成] を選択して、[S3 バケット名] を入力し、次に新しいバケットを作成する [リージョン] を選択します。

13. [バケットポリシー] を確認します。既存のバケットを選択する場合は、データエクスポートによって既存の S3 バケットポリシーが上書きされることを認識する必要があります。新しいポリシーでは、CUR とデータエクスポートの両方がエクスポートを配信できるようになります。
14. [S3 パスのプレフィックス] には、すべてのエクスポートデータを保存するために S3 バケットに作成されるディレクトリの名前を入力します。
15. タグでは、リソースを検索してフィルタリングしたり、 AWS コストを追跡したりするために、最大 50 個のタグを追加できます。

    > **Note**
    > タグの追加はオプションです。

16. [作成] を選択して、エクスポートの作成を完了します。

コストと使用状況ダッシュボードの作成

Amazon QuickSight によって提供される事前構築済みのコストと使用状況ダッシュボードをデプロイすることで、請求データとコスト管理データを視覚化できます。

コストと使用状況ダッシュボードを作成するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. [データエクスポート] ページで、[作成] または [コストと使用状況ダッシュボード] タイルのいずれかを選択します。
4. [作成] ページの [エクスポートタイプ] で、[QuickSight が提供するコストと使用状況ダッシュボード] を選択します。
5. [エクスポート名] に、ダッシュボードの名前を入力します。

   エクスポート名は、最大 128 文字で、一意である必要があります。有効な文字は、a-z、A-Z、0-9、 - (ハイフン)、および _ (アンダースコア) です。
6. [QuickSight ダッシュボード設定] では、[アカウント名]、[アカウント ID]、[アカウントエディション]、および、[認証方法] などの QuickSight アカウントの詳細が自動的に入力されます。

   1. QuickSight アカウントの詳細が自動的に入力されない場合は、QuickSight を初めて使用する場合は [アカウントを作成] を選択してサインアップするか、QuickSight の既存のお客様の場合は QuickSight アカウントにログインします。
   2. QuickSight アカウントを正常に作成またはログインすると、成功メッセージが表示されます。ウィンドウを閉じて、[データエクスポート] に戻ります。
   3. [QuickSight ダッシュボード設定] で、[更新] を選択します。

   > **Note**
   >
   > この機能には Enterprise Edition が必要です。
7. [QuickSight 名前空間] に、[名前空間] を入力します。
8. [QuickSight ユーザー名] には、QuickSight ダッシュボードにアクセスするアクセス許可を持つユーザーの詳細を入力します。

9. QuickSight リージョンで、QuickSight ダッシュボードを作成する AWS リージョンを選択します。
10. [データテーブルコンテンツ設定] と [データテーブル配信オプション] はあらかじめ設定されており、編集できません。
11. [データエクスポートストレージ設定] の [S3 バケット] 名で [設定] を選択します。
12. [S3 バケットの設定] ダイアログボックスで、次のいずれかを実行します。
    - 既存のバケットを選択します。
    - [バケットの作成] を選択して、[S3 バケット名] を入力し、次に新しいバケットを作成する [リージョン] を選択します。
13. [バケットポリシー] を確認し、[バケットの作成] を選択します。
14. [S3 パスのプレフィックス] に、エクスポート名の前に付加する S3 パスのプレフィックスを入力します。
15. [サービスアクセス]で、QuickSight を認可する方法を選択します。
    - 新しいサービスロールの作成 (デフォルト)
    - 既存のサービスロールを使用
16. タグでは、リソースを検索してフィルタリングしたり、 AWS コストを追跡したりするために、最大 50 個のタグを追加できます。

    > **Note**
    > タグの追加はオプションです。

17. [作成] を選択します。

AWS Billing and Cost Management コンソールのデータエクスポートページにいつでも戻り、コストと使用状況ダッシュボードが最後に更新された日時を確認できます。

レガシー CUR エクスポートの作成

従来のコストと使用状況レポート (CUR) のデータエクスポートを作成できます。このワークフローはレガシー `cur` API を使用しており、SQL を使用してエクスポートのコンテンツを定義することはできません。追加の列と SQL アクセスを備えた CUR 2.0 は、標準データエクスポートとしてのみ使用できます。

レガシーデータエクスポートを作成するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. [作成] を選択します。
4. [作成] ページの [エクスポートタイプ] で、[レガシー CUR エクスポート] を選択します。
5. [エクスポート名] に、エクスポートの名前を入力します。
6. [コンテンツをエクスポート] で、CUR エクスポートに含めるデータを選択します。
   - [追加エクスポートコンテンツ] で、[リソース ID を含める] を選択して各リソースの ID をエクスポートに含めます。

     > **Note**
     > リソース ID を含めると、リソースごとに個別の明細項目が作成されます。これにより、 AWS の使用状況に基づいて、エクスポートのサイズが大幅に増加する可能性があります。

   - 共有リソースの詳細なコストと使用状況を含めるには、[コスト配分データの分割] を選択します (Amazon ECS と Amazon EKS)。

     > **Note**
     > コスト配分データの分割を含めると、リソース (つまり ECS タスクと Kubernetes ポッド) ごとに、個別の明細項目が作成されます。これにより、 AWS 使用量に基づいてコストと使用状況レポートのサイズが大幅に増加する可能性があります。

   - [手動割引形式を有効にする] を選択すると、割引が標準の自動形式ではなく手動割引形式でコストと使用状況レポートに表示されるように変換されます。

     > **Note**
     > これは割引自動化を利用しているお客様のみ利用可能です。

7. [データテーブル配信オプション] の [時間粒度] で、次のいずれかを選択します。
   - [時間単位] エクスポートの明細項目を 1 時間ごとに集計する場合に選択します。

- [日単位] エクスポートの明細項目を 1 日ごとに集計する場合に選択します。
- [月単位] エクスポートの明細項目を 1 か月ごとに集計する場合に選択します。

8. [レポートバージョニング] では、次のいずれかを選択します。

   - [新しいレポートバージョンの作成]: 各レポートの更新は、同じ請求期間の配信であっても、別のディレクトリに書き込まれます。これを選択すると、長期間にわたるエクスポートの監査機能が向上します。
   - [既存のレポートを上書き]: レポートを更新するたびに、同じ請求期間内の前回の配信が上書きされます。新しい請求期間の配信は、新しいファイルおよびディレクトリとして配信されます。Amazon S3 のストレージコストを節約するために、これを選択します。

9. [レポートデータ統合] で、コストと使用状況レポートを Amazon Athena、Amazon Redshift、または Amazon QuickSight と統合できるようにするかを選択します。レポートは、以下の形式で圧縮されています。

   - [Amazon Athena]: Amazon Athena に最適な配信オプション (Parquet ファイル形式で、既存のレポートを上書きする) を選択します。また、統合のセットアップに使用できるスクリプトも配信されます。
   - [Amazon Redshift]: Amazon Redshift に最適な配信オプション (gzip/csv ファイル形式) を選択します。また、統合のセットアップに使用できるスクリプトも配信されます。
   - [Amazon QuickSight]: Amazon QuickSight に最適な配信オプション (gzip/csv ファイル形式) を選択します。

10. [圧縮タイプとファイル形式] では、次のいずれかを選択します。

    - Parquet − Parquet
    - gzip — text/csv
    - zip – text/csv

11. [データエクスポートストレージ設定] の [S3 バケット] 名で [設定] を選択します。
12. [S3 バケットの設定] ダイアログボックスで、次のいずれかを実行します。

    - 既存のバケットを選択します。
    - [バケットの作成] を選択して、[S3 バケット名] を入力し、次に新しいバケットを作成する [リージョン] を選択します。

13. [バケットポリシー] を確認し、[バケットの作成] を選択します。
14. [S3 パスのプレフィックス] に、エクスポート名の前に付加する S3 パスのプレフィックスを入力します。

15. タグでは、リソースを検索してフィルタリングしたり、 AWS コストを追跡したりするために、最大 50 個のタグを追加できます。

> **Note**
> タグの追加はオプションです。

16. [レポートを作成] を選択します。

データクエリ - SQL クエリとテーブル設定

データエクスポートでは、提供されているテーブル (CUR 2.0 など) に対して実行される SQL クエリ (列選択、行フィルター、列エイリアシング) を記述できます。各テーブルには、テーブルに含まれるデータを変更するテーブル設定がある場合もあります。例えば、CUR 2.0 では、時間単位、日単位、月単位の時間詳細度を選択する設定や、コストと使用状況のデータをリソースレベルの粒度で追加する設定を指定できます。

エクスポートデータクエリを完全に定義するには、次の 2 つの属性を指定する必要があります。

- SQL クエリ: SQL ステートメントはテーブルに対して実行され、エクスポートによって返されるデータを決定します。
- テーブル構成: テーブル構成設定により、SQL クエリがテーブルに対して実行される前に、テーブルに含まれるデータが変更されます。

[データエクスポート] コンソールページでは、選択内容に基づいて SQL ステートメントとテーブル設定を作成するワークフローを使用できます。データエクスポート SDK/CLI では、独自の SQL ステートメントとテーブル設定を記述できます。

データエクスポートの SQL ステートメント (`QueryStatement`) は次の構文を使用します。

```
SELECT <column_name_a>, <column_name_b>.<attribute_name> AS <new_name>, ...
FROM <TABLE_NAME>
[ WHERE <column_name> OPERATOR <value> AND|OR ... ]
[ LIMIT number ]
```

データエクスポートのテーブル設定 (`TableConfigurations`) は次の構文を使用します。

```
{"<TABLE_NAME>":
```

```
    {"<CONFIGURATION_NAME_A>": "<value>",
     "<CONFIGURATION_NAME_B>": "<value>",
     ...}
            }
```

SQL クエリ

SQL クエリ: SQL ステートメントはテーブルに対して実行され、エクスポートによって返されるデータを決定します。SQL ステートメントはエクスポートの作成後に変更できますが、選択したテーブルは変更できません。

SQL ステートメント (QueryStatement フィールド) には、最大 36,000 文字を使用できます。

データエクスポート SQL クエリで使用可能なキーワードは次のとおりです。

> **Note**
>
> キーワードでは大文字小文字を区別しません。列名、テーブル名では大文字と小文字が区別されます。

SELECT

必須。

テーブルから選択する列を指定します。1 つのクエリにつき 1 つの SELECT ステートメントのみ使用できます。

ドット演算子 `.` を使用して、MAP 列または STRUCT 列の属性を別の列として選択することを指定します。SQL 出力の結果となる列の名前は、デフォルトでは属性名です。

例えば、製品の MAP 列から属性を選択できます。

```
SELECT product.from_location FROM COST_AND_USAGE_REPORT
```

これにより、`product` 列から `from_location` 属性が選択され、その属性のデータを持つ新しい列が作成されます。デフォルトでは、出力でこの列の名前は `from_location` になります。ただし、名前は `AS` で変更できます。

各テーブルで使用可能な MAP 列と STRUCT 列、およびこれらの列の属性の詳細については、「データエクスポートテーブルディクショナリ」を参照してください。

AS

オプション。

選択されている列の名前を変更できるようにします。新しい列名には、スペースや英数字 (a-z、A-Z、0-9) およびアンダースコア (_) 以外の文字を使用することはできません。列のエイリアスを定義するときに、他の文字を使用するために引用符を使用することはできません。

エイリアシングは、MAP または STRUCT 列の属性を選択し、結果の列の名前を CUR のスキーマと一致させるために変更する場合に便利です。例えば、CUR の `product_from_location` 列の表示方法を一致させるには、CUR 2.0 テーブルを使用してデータエクスポートに次のクエリを記述します。

```
SELECT product.from_location AS product_from_location FROM
COST_AND_USAGE_REPORT
```

これにより、`product_from_location` という名前の列を含むエクスポートが作成されます。

FROM

必須。

クエリするテーブルを指定します。1 つのクエリにつき 1 つの FROM ステートメントのみ作成できます。

WHERE

オプション。

指定した句に一致する行のみをフィルタリングします。

WHERE 句では以下の演算子がサポートされます。

- = 値は文字列または数値と一致する必要があることを表します。
- != および <> 値は、指定した文字列または数値と一致しないことを表します。
- <、<=、> および >= 値は、数値より小さい、数値以下、数値より大きい、または数値以上であることを表します。
- AND は、指定された 2 つの条件がともに true である場合に一致することを表します。AND キーワードは、2 つ以上の条件を指定するために使用することもできます。
- OR は、指定された条件のいずれかが true である場合に一致することを表します。複数の OR キーワードを使用して、2 つ以上の条件を指定することができます。
- NOT 指定された条件が true でない場合に一致することを表します。

- IN キーワードの後の括弧内で指定された値のいずれかが true である場合に一致することを表します。
- 括弧を使用すると、複数条件の WHERE 句をコンストラクトできます。

> **Note**
>
> 文字列を演算子の後の値として表現する場合は、二重引用符の代わりに一重引用符 ' を使用します。一重引用符をエスケープする必要はありません。例えば、次の WHERE ステートメントを記述できます。
>
> `WHERE line_item_type = 'Discount' OR line_item_type = 'Usage'`

LIMIT

オプション。

クエリによって返される行の数を、指定した値に制限します。

テーブル設定

テーブル設定はユーザーが制御するプロパティで、ユーザーはデータエクスポートでクエリされる前にテーブルのデータまたはスキーマを変更するように設定できます。テーブル設定は JSON ステートメントとして保存され、 AWS SDK/CLI のユーザー入力またはコンソールのユーザー選択を通じて指定されます。

例えば、CUR 2.0 には、データの粒度 (時間単位、日単位、月単位)、リソースレベルの詳細なデータを含めるかどうか、分割コスト配分データを含めるかどうかを変更するテーブル設定があります。すべてのテーブルに設定があるわけではありません。各テーブルで使用できる設定の詳細については、「データエクスポートテーブルディクショナリ」を参照してください。

各テーブル設定パラメータには、ユーザーがテーブル設定を指定しなかった場合に想定されるデフォルト値があります。テーブル設定はエクスポートの作成後に変更することはできません。

データエクスポートの表示と管理

エクスポートの詳細を表示するには、 AWS Billing and Cost Management コンソールの [データエクスポート] ページを使用します。エクスポートファイルを表示するには、[データエクスポート] ページの Amazon S3 バケットの S3 コンソールリンクを使用します。エクスポートダッシュボードを表示するには、[データエクスポート] ページの QuickSight リンクを使用するか、QuickSight コンソールに直接アクセスしてダッシュボードを探します。

エクスポートの詳細、ファイル、およびダッシュボードを表示するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. [エクスポートとダッシュボード] リストで、表示したいエクスポートの名前を探します。
4. エクスポートの詳細を表示するには、[エクスポート名] 列のリンクを選択し、エクスポート設定を説明する概要ページを表示します。
5. エクスポートファイルを表示するには、[S3 バケット] 列のリンクを選択して、バケットの S3 コンソールに移動します。
6. QuickSight ダッシュボードを表示するには、[エクスポートタイプ] 列の [コストと使用状況ダッシュボード] リンクを選択します。

[エクスポートとダッシュボード] リストの列の概要は次のとおりです。

- [エクスポート名]: エクスポートを作成したときに選択した名前。
- [ステータス]: エクスポートの状態。2 つの値があります。
  - [正常]: このステータスは、直近のエクスポート配信が成功したことを示します。

    > **Note**
    >
    > すべてのデータがダッシュボードに入力されるまでに最大 24 時間かかることがあるため、コストと使用状況ダッシュボードに当月のデータを表示できない場合があります。エクスポートステータスが「正常」の場合は、ダッシュボードが当月のデータで更新されるまで 24 時間かかります。

  - [異常]: このステータスは、直近のエクスポート配信が失敗したことを示します。

- [エクスポートタイプ]: 作成されたエクスポートのタイプ。データエクスポートには次の 3 種類のエクスポートがあります。
  - [標準データエクスポート]: Amazon S3 に定期的に配信されるテーブルのカスタマイズされたエクスポート。
  - [コストと使用状況ダッシュボード]: Amazon QuickSight へのエクスポートと統合で、事前に構築されたコストと使用状況のダッシュボードを展開します。これがダッシュボードへのリンクになります。
  - [レガシー CUR エクスポート]: レガシーコストおよび使用状況レポート (CUR) のエクスポートです。
- [データテーブル]: エクスポートによってクエリされるテーブル。
- 作成日: エクスポートが作成された日時。
- 最終更新日: エクスポートが最後に更新された日時。
- [S3 バケット]: エクスポート先の S3 バケット。

エクスポート配信について

次のセクションでは、エクスポート配信について説明します。

- エクスポート S3 の親ディレクトリ構造: エクスポートが配信される S3 ディレクトリで、エクスポートデータがどのように構造化されるか。
- エクスポートの更新: S3 ディレクトリでエクスポートが更新される頻度。
- エクスポートの上書きと新規作成: エクスポート配信が上書きによってどのように変化し、新しい配信設定が作成されるか。
- エクスポートデータのファイル名とチャンク: エクスポートファイル (gzip/csv または Parquet) の名前の付け方。

エクスポート S3 の親ディレクトリ構造

各エクスポートは、クエリからのデータが (1 つ以上の gzip/csv ファイルまたは Parquet ファイルとして) S3 に配信され、エクスポートが実行された時点でのエクスポート定義に関する情報を含む `Manifest.json` メタデータファイルも送信されます。

[データ]

エクスポートクエリの結果のデータは、次の S3 ファイルパスに保存されます。

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/
```

パーティションはクエリ対象のテーブルに対応しています。CUR 2.0 の場合、パーティションは特定の CUR 2.0 エクスポートの「請求期間」に対応します。

`prefix`: エクスポートに割り当てる S3 ファイルプレフィックス。

`export-name`: エクスポートに割り当てる名前。

`partition`: パーティションは、1 つのテーブルを配信用に別々のテーブルに分割する方法を示しています。CUR 2.0 では、パーティションは `BILLING_PERIOD=YYYY-MM` のフォーマットで「請求期間」に対応します。例えば、2023 年 11 月のパーティションは 2023-11 です。

S3 ファイルパスの例を次に示します。

```
s3://my-data-export-s3-bucket/my-cur-files/business_group_a_cur/data/
BILLING_PERIOD=2023-11
```

メタデータ

クエリの `Manifest.json` メタデータファイルは、次の S3 ファイルパスに保存されます。

```
s3://<bucket-name>/<prefix>/<export-name>/metadata/<partition>/<export-
name>-Manifest.json
```

`Manifest.json` ファイルは、エクスポートが更新されるたびに更新されます。エクスポートによって新しいパーティションが作成されるたびに、新しい `Manifest.json` ファイルが作成されます。CUR 2.0 では、新しい請求期間が始まると新しい `Manifest.json` ファイルが生成されます。

マニフェストファイルには、次の情報が含まれています。

- エクスポートに含まれるすべての列。
- エクスポートファイルとそのファイルパスのリスト。このリストをプログラムで読み込み、取り込むファイルを特定することをおすすめします。
- エクスポートの対象となる期間。

`Manifest.json` は、すべてのエクスポートデータファイルが S3 に配信された後にのみ配信されます。

エクスポートの更新

データエクスポートは、ソースデータが更新されるたびにエクスポートを更新します。CUR 2.0の場合、少なくとも 1 日に 1 回実行されます。現在の請求期間 (パーティション) は請求期間が終了するまで更新され、その時点で次の請求期間の配信が開始されます。次の請求期間の配信には、その請求期間の料金と請求データのみが含まれます。請求期間が終了したら、終了してから最初の 2 週間以内に、以前の請求期間のエクスポート配信を更新 AWS できます。

エクスポートの上書きと新規作成

エクスポートを作成する場合、新しいエクスポートファイルを作成するか、更新ごとに既存のエクスポートファイルを上書きするかを選択できます。

新規作成

新しいエクスポートファイルを作成すると、エクスポートの更新がすべて保持されるため、S3 ストレージの使用量が増えます。以前のエクスポートファイルを上書きすると、各請求期間の更新の最新バージョンのみが保持されるため、S3 ストレージの使用量が少なくなります。

「新規作成」モードでは、エクスポートファイルは次の S3 パスに配信されます。

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/<timestamp>-
<execution-id>
```

`timestamp` はエクスポートが実行された日時です。`execution-id` は実行に割り当てられた一意の ID です。

「新規作成」では、エクスポートを実行するたびに 2 つの `Manifest.json` ファイルが配信されます。1 つは `metadata/<partition>/<timestamp>-<execution-id>` ディレクトリに保存され、もう 1 つは `metadata/<partition>` ディレクトリで上書きされます。`metadata/<partition>` ディレクトリ内のマニフェストは常に最新の更新内容を表し、そのデータを使用して最近更新されたエクスポートファイルの場所を識別します。

上書き

上書きは、同じパーティション (つまり請求期間) の更新にのみ適用されます。新しい請求期間が始まると、エクスポートは最新のパーティションまたは請求期間に基づいた名前で新しい S3 ディレクトリを作成し、そこに新しいエクスポートパーティションの配信を開始します。特定のパーティションのデータが更新されない限り、前のパーティションのエクスポートは上書きされません。

「上書き」モードでは、エクスポートファイルは次の S3 パスに配信されます。

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/
```

このファイルディレクトリ内のエクスポートファイルは、同じパーティション (つまり、請求期間) が配信されるたびに上書きされます。

エクスポートのサイズが十分になると、エクスポートファイルは複数の「チャンク」(個別の gzip/csv ファイルまたは Parquet ファイル) として配信されます。その月の間にエクスポートのサイズが減少した場合 (クエリの変更やデータの修正により)、エクスポートを更新するのに必要なチャンクが少なくなる可能性があります。この場合、データエクスポートは前回の更新で発生した余分なチャンクを空のデータで上書きします。

上書きの場合、エクスポートを実行するたびに 1 つの `Manifest.json` ファイルが配信されます。ファイルは `metadata/<partition>` ディレクトリに保存され、更新されるたびに上書きされます。

エクスポートデータのファイル名とチャンク

エクスポートは、1 回の実行結果を 1 つのファイル (gzip/csv または Parquet) として配信するか、エクスポートのサイズが十分になったときに複数の「チャンク」(個別の gzip/csv または Parquet ファイル) として配信します。

gzip/csv ファイル形式のエクスポートの名前は次のとおりです。

```
<export-name>-<chunk-number>.csv.gz
```

Parquet 形式のエクスポートの名前は次のとおりです。

```
<export-name>-<chunk-number>.snappy.parquet
```

チャンク番号は常に 5 桁です。チャンク番号は `00001` から順に列挙されます。

概要

新規作成用のディレクトリを含むエクスポートデータファイル名

Parquet:

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/<timestamp>-
<execution-id>/<export-name>-<chunk-number>.snappy.parquet
```

gzip/csv:

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/<timestamp>-
<execution-id>/<export-name>-<chunk-number>.csv.gz
```

上書き用のディレクトリを含むエクスポートデータファイル名

Parquet:

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/<export-
name>-<chunk-number>.snappy.parquet
```

gzip/csv:

```
s3://<bucket-name>/<prefix>/<export-name>/data/<partition>/<export-
name>-<chunk-number>.csv.gz
```

新規作成用のディレクトリを含むマニフェストファイル名

「新規作成」モードは `Manifest.json` を 2 つの場所に配信します。

1 つ目の場所は、エクスポートの特定の実行を表すフォルダ (`timestamp` および `execution-id` という名前) にあります。このマニフェストはその特定の実行に対応しています。ファイルパスは次のとおりです。

```
s3://<bucket-name>/<prefix>/<export-name>/metadata/<partition>/
<timestamp>-<execution-id>
```

2 つ目の場所は、すべての実行内容を含むパーティションフォルダです。このマニフェストは、最後にエクスポートを実行したときと同じファイルです。このマニフェストを読むと、最近のすべてのエクスポートファイルの正確なファイルパスを確認できます。ファイルパスは次のとおりです。

```
s3://<bucket-name>/<prefix>/<export-name>/metadata/<partition>/
Manifest.json
```

上書き用のディレクトリを含むマニフェストファイル名

「上書き」モードでは `Manifest.json` を 1 つの場所に配信します。

```
s3://<bucket-name>/<prefix>/<export-name>/metadata/<partition>
```

このディレクトリのマニフェストは、特定のパーティション (つまり、請求期間) が更新されるたびに上書きされます。

エクスポートの詳細の編集

AWS Billing and Cost Management コンソールのデータエクスポートページを使用して、エクスポートの詳細を編集できます。

エクスポートの詳細を編集するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. エクスポートのリストから、編集するエクスポートの名前を選択します。
4. [エクスポートの詳細] ページで、[編集] を選択します。
5. [編集] モードでは、エクスポートの列の選択とエクスポートの上書き設定を更新できます。

   データテーブル、追加のエクスポートコンテンツ (テーブル設定)、または既存のエクスポートの圧縮タイプとファイル形式は更新できません。これらの詳細を変更するには、新しいエクスポートを作成する必要があります。

エクスポートタグの編集

AWS Billing and Cost Management コンソールのデータエクスポートページを使用して、エクスポートタグを編集できます。

エクスポートタグを編集するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. エクスポートのリストから、編集するエクスポートの名前を選択します。
4. ページ下部の [タグ] セクションで、[タグを管理] を選択します。
5. [タグを管理] ページで、次の操作を実行します。

   - タグを追加するには、[新しいタグを追加] を選択します。最大 50 個のタグを追加して、リソースを検索およびフィルタリングしたり、 AWS コストを追跡したりできます。
   - 既存のタグを削除するには、削除するエントリの横にある [削除] を選択します。
6. [保存] を選択します。

エクスポートの削除

AWS Billing and Cost Management コンソールのデータエクスポートページを使用して、エクスポートを削除できます。

エクスポートを削除するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [データエクスポート] を選択します。
3. エクスポートのリストから、削除するエクスポートの名前を選択します。
4. [エクスポートの詳細] ページで、[削除] を選択します。
5. もう一度 [削除] を選択し、エクスポートを削除することを確認します。

> **Note**
>
> この手順により、データエクスポートからエクスポートが削除されます。ただし、Amazon S3 バケットに保存されているオブジェクトは削除されません。
> コストと使用状況ダッシュボードの場合、上記の手順はコストと使用状況ダッシュボードをデータエクスポートから削除します。ただし、S3 バケット、QuickSight ダッシュボード、およびその他の QuickSight リソースに保存されているオブジェクトは削除されません。QuickSight からコストと使用状況ダッシュボードを削除するには、「Amazon QuickSight ダッシュボードの削除」を参照してください。
> Amazon QuickSight ダッシュボードを削除すると、ダッシュボードは、アカウントから、およびダッシュボードが属していたすべてのフォルダから完全に削除されます。削除されたダッシュボードにはアクセスできなくなります。所有しているか共同所有しているダッシュボードのみを削除できます。

AWS Organizations でのデータエクスポートの使用

データエクスポートは AWS Organizations と連携して、管理アカウントが組織内のすべてのアカウントのデータを含むエクスポートを生成できるようにします。メンバーアカウントでもデータのエクスポートを作成できますが、これらのエクスポートに含まれるのは、その特定のメンバーアカウントの請求およびコスト管理データのみです。管理アカウントがすべてのメンバーアカウントのデータを受信するかどうかを制御する設定は、データエクスポートテーブルによって異なります。各テーブル

にメンバーアカウントデータを含めるかどうかを決定する方法については、以下のセクションを参照してください。

- コストと使用状況レポート 2.0 (CUR 2.0)
- コスト最適化の推奨事項 (Cost Optimization Hub から)
- AWS 列を含む FOCUS 1.0
- コストと使用状況ダッシュボード
- 炭素排出量

エクスポートの作成を許可または制限する IAM ポリシーは、管理アカウントとメンバーアカウントの両方で同じです。

AWS Organizations 管理アカウントの管理者であり、メンバーアカウントにエクスポートを作成させたくない場合は、メンバーアカウントがエクスポートを作成できないようにするサービスコントロールポリシー (SCP) を適用できます。SCP では、メンバーアカウントによる新しいエクスポートの作成を防止できますが、以前に作成されたエクスポートは削除されません。

> **Note**
>
> SCP はメンバーアカウントにのみ適用されます。管理アカウントがエクスポートを作成できないようにするには、管理アカウントのユーザーロールに添付されている IAM ポリシーを変更します。

データエクスポートテーブルディクショナリ

データエクスポートには、エクスポートの作成時にクエリできるテーブルがいくつか用意されています。

次のトピックでは、各列の定義と共に、使用可能な各テーブルのスキーマについて説明します。

トピック

- コストと使用状況レポート (CUR) 2.0
- コスト最適化の推奨事項 (Cost Optimization Hub から)
- AWS 列を含む FOCUS 1.0
- AWS 列を含む FOCUS 1.0 (プレビュー)
- コストと使用状況ダッシュボード
- 炭素排出量

コストと使用状況レポート (CUR) 2.0

CUR 2.0 テーブルは、コストと使用状況レポート (CUR) と同じ情報を提供しますが、いくつか改善されています。

コストと使用状況レポート 2.0 では、コストと使用状況レポートに比べて以下の点が改善されています。

- 整合性のあるスキーマ: CUR 2.0 には固定列セットが含まれていますが、CUR に含まれる列は、AWS サービス、コストカテゴリ、リソースタグの使用に応じて毎月異なる場合があります。
- ネストされたデータ: CUR 2.0 では、CUR の特定の列を、折りたたまれた列のキーと値のペアを持つ個々の列に折りたたむことで、データのスパース性を軽減します。ネストされたキーは、元の CUR スキーマとデータと一致するように、オプションで個別の列としてデータエクスポートでクエリできます。
- 追加列: CUR 2.0 には、bill_payer_account_name と line_item_usage_account_name という 2 つの列が追加されています。

CUR 2.0 の SQL テーブル名は `COST_AND_USAGE_REPORT` です。

テーブル設定

テーブル設定はユーザーが制御するプロパティで、ユーザーはデータエクスポートでクエリされる前にテーブルのデータまたはスキーマを変更するように設定できます。テーブル設定は JSON ステートメントとして保存され、 AWS SDK/CLI でのユーザー入力またはコンソールでのユーザー選択によって指定されます。

CUR 2.0 には以下のテーブル設定があります。

設定名	説明	有効値
TIME_GRANULARITY	この設定により、CUR 2.0 テーブルのコストと使用量の明細項目が、異なる時間精度を持つように変更されます。 例えば、[時間単位] を選択すると、すべての明細項目が 1 時間の使用量を表すようになります。	HOURLY, DAILY, MONTHLY
INCLUDE_RESOURCES	この設定では、CUR 2.0 テーブルのコストと使用量の明細項目がリソースレベルの細分化に変更され、テーブルスキーマに「line_item_resource_id」列が追加されます。 この設定を有効にすると、CUR 2.0 テーブルには、そのサービスの合計使用量ではなく、そのサービスで使用されたリソースごとに 1 つの明細項目が表示されます。 この設定を有効にすると、エクスポートの行数が大幅に増	TRUE、FALSE

設定名	説明	有効値
	加し、ファイルサイズも大きくなる可能性があります。	
INCLUDE_SPLIT_COST_ALLOCATION_DATA	この設定により、分割コスト配分データと列 (split_line_item_*) が CUR 2.0 テーブルに追加されます。このデータは、特定の AWS リソースの使用量をさまざまなビジネスユニットまたはチームにどのように割り当てることができるかを示します。 この設定を有効にすると、EC2 インスタンスをそのインスタンスで実行されているさまざまなコンテナにどのように割り当てることができるかを示す行と列を追加できます。詳細については、「分割コスト配分データについて」を参照してください。	TRUE、FALSE

設定名	説明	有効値
INCLUDE_MANUAL_DISCOUNT_COMPATIBILITY	注意: この設定は、割引が自動的に計算される割引自動化プログラムに登録した AWS のお客様にのみ適用されます。 この設定では、CUR 2.0 テーブル内の割引が CUR に「手動」で追加されたときのように (通常は個別の明細項目として) 表示されるように変更され、スキーマから 2 つの列 (「discount」と「total_discount」) が削除されます。	TRUE、FALSE

AWS Organizations のサポート

CUR 2.0 テーブルは、 AWS Organizations の統合請求機能で行った設定を継承します。一括決済が有効になっている場合、管理アカウントとメンバーアカウントにはさまざまな動作があります。管理アカウントを使用している場合、CUR 2.0 テーブルには、管理アカウントと組織内のすべてのメンバーアカウントのコストと使用状況データが含まれます。メンバーアカウントを使用している場合、CUR 2.0 テーブルには、そのメンバーアカウントのコストと使用状況データのみが含まれます。

組織に加入すると、メンバーアカウントは、そのアカウントが組織のメンバーであった期間のデータのみをエクスポートできます。例えば、あるメンバーアカウントが組織 A を離れ、同月 15 日に組織 B に加入したとします。その後、メンバーアカウントはエクスポートを作成します。メンバーアカウントは組織 B に加入した後にエクスポートを作成したため、その月のメンバーアカウントの CUR 2.0 のエクスポートには、そのアカウントが組織 B のメンバーであった期間のコストと使用状況データのみが含まれます。

メンバーアカウントが新しい組織に加入すると、そのメンバーアカウントのコストと使用状況データが新しい組織のエクスポートに記録されます。これは、管理アカウントをメンバーアカウントに変換して新しい組織に加入した場合と同じ結果です。

メンバーアカウントが組織を離れたり、スタンドアロンアカウントに変換された場合でも、メンバーアカウントは、以前のエクスポートが保存されている Amazon S3 バケットへのアクセス権限を持っている限り、以前のエクスポートにアクセスできます。

詳細については、「 Billing AWS User Guide」の「Consolidated Billing for AWS Organizations」を参照してください。

CUR 2.0 の列グループ

CUR 2.0 テーブルには 125 個の列があり、次のようにグループ化されています。

- 請求書: 請求期間の請求に関するデータ。
- コストカテゴリ: 明細項目に適用されるコストカテゴリに関するデータ。
- 割引: 適用されている割引に関するデータ。
- ID: 明細項目を識別するためのデータ。
- 明細項目: コスト、使用量、使用タイプ、価格レート、製品名などに関するデータ。
- 料金: 明細項目の料金に関するデータ。
- 製品: 明細項目で請求されている製品に関するデータ。
- 予約: 明細項目に適用される予約に関するデータ。
- リソースタグ: 明細項目に適用されるリソースタグに関するデータ。
- Savings Plans: 明細項目に適用される Savings Plans に関するデータ。
- 分割明細項目: 別の明細項目への分割コスト配分に関するデータ。

請求書列

請求書列には、請求期間の請求に関するデータが含まれます。

列名	説明	データ型
bill_bill_type	このレポートの対象となる請求書の種類。請求書の種類は 3 つあります。 •	string

列名	説明	データ型
	アニバーサリー: その月中に使用したサービスの明細項目。 • 購入: 前払いサービス料金の明細項目。 • 返金: 返金の明細項目。	
bill_billing_entity	請求書または取引が AWS Marketplace 向けか、他の AWS サービスの購入向けかを特定するのに役立ちます。	string
bill_billing_period_end_date	このレポートの対象となる請求期間の終了日 (UTC)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。	timestamp
bill_billing_period_start_date	このレポートの対象となる請求期間の開始日 (UTC)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。	timestamp
bill_invoice_id	特定の明細項目に関連付けられた ID。レポートが確定されるまでは、`[InvoiceId ]` が空白になります。	string
bill_invoicing_entity	請求書を発行する AWS エンティティ。	string

列名	説明	データ型
bill_payer_account_id	支払いアカウントのアカウント ID。 AWS の組織の場合、これは管理アカウントのアカウント ID です。	string
bill_payer_account_name	支払いアカウントのアカウント名。 AWS の組織の場合、これは管理アカウントのアカウントの名前です。	string

コストカテゴリ列

コストカテゴリ列には、明細項目に適用されるコストカテゴリに関するデータが含まれます。

列名	説明	データ型
cost_category	特定の明細項目のコストカテゴリのキーと値のペアとその値を含むマップ列。これらのキーと値は、コストカテゴリ機能で作成した分類ルールに基づいて入力されます。 コストカテゴリキーは、その値が特定の明細項目に適用される場合にのみマップ列に表示されます。 この列のキーは、ドット演算子を使用して個別の列としてクエリできます。詳細については、「データクエリ」を参照してください。	map <string, string>

割引列

割引には適用されている割引に関するデータが含まれています。

列名	説明	データ型
discount	テーブル設定: 削除: [INCLUDE MANUAL DISCOUNT COMPATIBILITY] この明細項目に適用される特定の割引のキーと値のペアを含む「構造体」列。キーは割引タイプに対応し、値は割引値またはその他の情報に対応します。この列の値は、特定のキーに応じて「数値」または「文字列」のデータ型になります。 この列のキーは、ドット演算子を使用して個別の列としてクエリできます。詳細については、「データクエリ」を参照してください。 [MANUAL DISCOUNT COMPATIBILITY] が有効になっている場合、この列は使用できません。有効にすると、割引はこの列ではなく個別の明細項目として入力されます。	map <string, double>
discount_bundled_discount	明細項目に適用されるバンドル割引。バンドル割引は、別のサービスまたは機能の使用	double

列名	説明	データ型
	量に基づいて、サービスまたは機能の無料または割引の使用を提供する使用量ベースの割引です。バンドル割引の例は次のとおりです。 • AWS Shield Advanced を使用する場合、 AWS WAF に対して別途料金を支払う必要はありません。AWS WAF の使用は Shield Advanced AWS にバンドルされています。Shield Advanced の詳細については、「Amazon CloudFront AWS の料金」を参照してください。 Amazon CloudFront • AWS Network Firewall を使用して NAT ゲートウェイを作成する場合、標準の NAT ゲートウェイ処理と 1 時間あたりの使用料金は、GB および使用時間あたりのファイアウォール処理で one-to-one ベースで免除されます。詳細については、「AWS Network Firewall の料金」を参照してください。 • Amazon インタラクティブビデオサービス (IVS) チャットでは、ビデオ入力	

列名	説明	データ型
	が 1 時間送信されるごとに、2,700 通のメッセージが送信され、27 万件のメッセージが配信されます。追加料金は発生しません。詳細については、「Amazon Service Video Service 料金」を参照してください。	
discount_total_discount	テーブル設定: 削除: [INCLUDE MANUAL DISCOUNT COMPATIBILITY] 対応する明細項目のすべての discount 列の合計。 [MANUAL DISCOUNT COMPATIBILITY] が有効になっている場合、この列は使用できません。有効にすると、割引はこの列ではなく個別の明細項目として入力されます。	double

アイデンティティ列

アイデンティティ列には明細項目を識別するためのデータが含まれます。

列名	説明	データ型
identity_line_item_id	このフィールドは明細項目ごとに生成され、特定のパーティション内で一意です。これは、CUR の配信全体 (つ	string

列名	説明	データ型
	まり、更新内のすべてのパーティション) AWS でフィールドが一意であることを保証するものではありません。明細項目 ID は、異なるコストと使用状況レポート間で一貫性がないため、異なるレポート間の同じ明細項目を識別するために使用することはできません。	
identity_time_interval	この明細項目が適用される時間の間隔、以下の形式: `YYYY-MM-DDTHH:mm:ssZ/YYYY-MM-DDTHH:mm:ssZ` 。時間の間隔は UTC で表され、レポートの詳細度に応じて毎日または毎時のいずれかになります。	string

明細項目列

明細項目列にはコスト、使用量、使用タイプ、価格レート、製品名などに関するデータが含まれます。

列名	説明	データ型
line_item_usage_account_name	この明細項目を使用したアカウントの名前。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。このフィールドを使用して、アカウント別	string

列名	説明	データ型
	のコストや使用状況を追跡することができます。	
line_item_availability_zone	この明細項目をホストするアベイラビリティーゾーン。例えば、`us-east-1a` 、`us-east-1b` です。	string
line_item_blended_cost	`BlendedRate` に `UsageAmount` を乗じた値。 [BlendedCost] は、[LineItemType] が [Discount] である明細項目については空白になります。ディスカウントは、メンバーアカウントと SKU で集計されたメンバーアカウントの非ブレンドコストのみを使用して計算されます。そのため、[BlendedCost] はディスカウントに使用できません。	double

列名	説明	データ型
line_item_blended_rate	`BlendedRate` は、組織全体で各 SKU に発生した平均コストです。 例えば、Amazon S3 ブレンドレートは、ストレージの合計コストを、1 か月間に保存されたデータ量で割ったものです。RI を持つアカウントの場合、ブレンドレートは RI とオンデマンドインスタンスの平均コストとして計算されます。 ブレンドレートは管理アカウントレベルで計算され、各メンバーアカウントにコストを割り当てるために使用されます。詳細については、「AWS 請求ユーザーガイド」の「ブレンドレートとコスト」を参照してください。	string
line_item_currency_code	この明細項目の通貨。すべての AWS お客様は、デフォルトで米ドルで請求されます。請求用の通貨を変更するには、「AWS 請求ユーザーガイド」の「請求の支払いに使用する通貨の変更」を参照してください。	string

列名	説明	データ型
line_item_legal_entity	特定の製品またはサービスの登録販売者。ほとんどの場合、請求書を発行する事業体と法律上の事業体は同じです。サードパーティーの AWS Marketplace トランザクションでは、値が異なる場合があります。可能な値は以下のとおりです: • Amazon Web Services, Inc.: AWS のサービスを販売する事業体。 • Amazon Web Services India Private Limited: インドの AWS サービスのリセラーとして機能するインドのローカルエンティティ。	string

列名	説明	データ型
line_item_line_item_description	明細項目タイプの説明。例えば、使用料の明細項目は、特定の期間に発生した使用タイプを要約したものです。 サイズ柔軟な RI の場合、説明は、利点が適用された RI に対応します。例えば、明細項目が `t2.micro` に対応し、`t2.small` RI がその使用に適用された場合、line_item_line_item_description には `t2.small` と表示されます。 RI ディスカウントが適用された使用料の明細項目の説明には、明細項目の対象となる料金プランが含まれています。	string

列名	説明	データ型
line_item_line_item_type	この明細項目の対象となる請求の種類。割り当てられるタイプは次のとおりです。 • BundledDiscount: 別のサービスまたは機能の使用量に基づいて、サービスまたは機能の無料または discount 利用を提供する使用量ベースの割引です。 • クレジット: 請求書 AWS に適用されたクレジット。が請求書を確定した翌月にアカウントにクレジット AWS を適用する場合、確定後にレポート AWS を更新するには、説明列を参照してください。 • 割引: の使用 AWS に適用されたすべての割引。この特定の詳細項目名は異なる場合があり、割引に基づいて解析する必要があります。詳細については、[lineItem/LineItemDescription] 列を参照してください。 • DiscountedUsage: リザーブドインスタンス (RI) の利点があるすべてのインスタンスの割合。 •	string

列名	説明	データ型
	Fee: サブスクリプションに支払った前払い年間料金。たとえば、[全前払い RI] または [一部前払い RI)] として支払った前払い金が該当します。 • 返金: が返金 AWS した金額のマイナスの料金。が請求確定後の月の返金をアカウント AWS に適用した場合、確定後にレポート AWS を更新する可能性がある詳細については、説明列を確認してください。 • RIFee: サブスクリプションの定期的な月額料金。たとえば、毎月支払う [一部前払い RI]、[前払いなし RI]、[全額前払い] の定期的な料金を指します。RIFee は全額前払い予約が 0 USD になる場合がありますが、reservation/AmortizedUpfrontFeeForBillingPeriod や reservation/ReservationARN などの他の列を提供するために、この行はこれらの予約タイプに対しても事前入力されます。 •	

列名	説明	データ型
	Tax: AWS が請求書に適用した税金。VAT、米国の消費税などが該当します。 • Usage: オンデマンドインスタンスレートで課金される使用料金。 • SavingsPlanUpfrontFee: 全額前払いまたは一部前払い Savings Plan を購入した場合のワンタイム前払い料金。 • SavingsPlanRecurringFee: 前払いなしまたは一部前払い Savings Plan に対応する毎時間の繰り返し課金。Savings Plan 定期料金は、前払いなしまたは一部前払い Savings Plan を購入した日に、最初に請求書に追加されます。初回購入後、 は各請求期間の最初の日に定期的な料金 AWS を追加します。 • SavingsPlanCoveredUsage: Savings Plan の対象となる任意のオンデマンドコスト。Savings Plan の対象となる使用量明細項目は、対応する Savings Plan 否定項目によって相殺されます。	

列名	説明	データ型
	• [SavingsPlanNegation] – 対応する Savings Plan の対象となる使用量項目に関連付けられている Savings Plan 特典による相殺コスト。	
line_item_net_unblended_cost	品目に対して支払っている実際のディスカウント後の費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
line_item_net_unblended_rate	その明細項目に対して支払っている実ディスカウント後レート。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	string

列名	説明	データ型
line_item_normalization_factor	インスタンスがテナンシーを共有している限り、 はインスタンスファミリーと AWS リージョンのすべてのインスタンスサイズにすべてのリージョン Linux または Unix Amazon EC2 および Amazon RDS RI 割引を適用 AWS できます。これは、組織内のメンバーアカウントの RI ディスカウントにも適用されます。Amazon EC2 および Amazon RDS のすべての新規および既存のサイズ柔軟な RI のサイズは、インスタンスサイズに基づいた正規化係数により決定されます。	double
line_item_normalized_usage_amount	サイズ柔軟な RI のために発生した使用量。正規化単位。[NormalizedUsageAmount] は、[UsageAmount] に [NormalizationFactor] を乗算したものに等しくなります。	double
line_item_operation	この明細項目の対象となる特定の AWS オペレーション。これは、明細項目の特定の使用状況を表します。たとえば、`RunInstances` の値は Amazon EC2 インスタンスのオペレーションを示します。	string

列名	説明	データ型
line_item_product_code	測定された製品のコード。例えば、Amazon EC2 は、Amazon Elastic Compute Cloud の製品コードです。	string
line_item_resource_id	テーブル設定: 追加理由: INCLUDE RESOURCES (オプション) レポートに個々のリソース ID を含めることを選択した場合、この列にはプロビジョンしたリソースの ID が表示されます。例えば、Amazon S3 ストレージバケット、Amazon EC2 コンピューティングインスタンス、または Amazon RDS データベースは、すべてリソース ID を持つことができます。このフィールドは、インスタンス化されたホストに関連付けられていない使用タイプ (データ転送、API リクエストなど) と明細項目タイプ (ディスカウント、クレジット、税金など) では空白です。	string
line_item_tax_type	この明細項目 AWS に適用された税金のタイプ。	string

列名	説明	データ型
line_item_unblended_cost	`UnblendedCost` は、`UnblendedRate` に `UsageAmount` を乗じた値です。	double
line_item_unblended_rate	AWS Organizations を使用するアカウントの一括請求では、非ブレンドレートは個々のアカウントのサービス使用量に関連するレートです。 RI 割引が適用されている Amazon EC2 と Amazon RDS 明細項目の場合、`Unblended Rate` は `0` です。RI ディスカウントが適用されている明細項目は、`LineItemType` が `DiscountedUsage` です。	string
line_item_usage_account_id	この明細項目を使用したアカウントの ID。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。このフィールドを使用して、アカウント別のコストや使用状況を追跡することができます。	string

列名	説明	データ型
line_item_usage_amount	指定した期間に発生した使用量。サイズ柔軟なリザーブドインスタンスについては、代わりに reservation/TotalReservedUnits 列を使用します。 特定のサブスクリプション料金では、UsageAmount が `0` になります。	double
line_item_usage_end_date	対応する明細項目の終了日時は UTC で表されます (その時刻は含まない)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。	timestamp
line_item_usage_start_date	明細項目の開始日時は UTC で表されます (その時刻を含む)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。	timestamp
line_item_usage_type	この明細項目の使用状況の詳細。例えば、`USW2-BoxUsage:m2.2xlarge` は米国西部 (オレゴン) リージョンの M2 ハイメモリダブルエクストララージインスタンスについて説明します。	string

料金列

料金列には明細項目の料金に関するデータが含まれます。

列名	説明	データ型
pricing_currency	料金データが表示される通貨。	string
pricing_lease_contract_length	RI の予約期間。	string
pricing_offering_class	リザーブドインスタンスの提供クラス。	string
pricing_public_on_demand_cost	パブリックオンデマンドインスタンスレートに基づく明細項目の合計コスト。複数のオンデマンドパブリックコストが発生する SKU がある場合は、最も高い階層の同等のコストが表示されます。無料利用枠や階層型価格を提供するサービスなどが該当します。	double
pricing_public_on_demand_rate	特定の使用料の明細項目に対する、この請求期間のパブリックオンデマンドインスタンスレート。複数のオンデマンドパブリック料金が発生する SKU がある場合、最も高い階層の同等の料金が表示されます。無料利用枠や階層型価格を提供するサービスなどが該当します。	string
pricing_purchase_option	この明細項目に選択した支払い方法。有効な値は、`All Upfront`、`Partial`	string

列名	説明	データ型
	Upfront、No Upfront です。	
pricing_rate_code	製品/オファー/価格-枠の組み合わせの一意のコード。製品および期間の組み合わせは、無料利用枠、低使用利用枠、高使用利用枠など、複数の価格ディメンションを持つことができます。	string
pricing_rate_id	明細項目のレートの ID。	string
pricing_term	AWS 使用量がリザーブドかオンデマンドか。	string
pricing_unit	使用コストの計算 AWS に使用される料金単位。例えば、Amazon EC2 インスタンスの使用料の料金単位は時間数です。	string

製品列

製品列には明細項目で請求されている製品に関するデータが含まれます。

列名	説明	データ型
product	特定の明細項目の複数の製品属性とその値のキーと値の組み合わせを含むマップ列。	map <string, string>

列名	説明	データ型
	製品属性は、その値が特定の明細項目に適用される場合にのみマップ列に表示されます。 Note レガシー CUR に表示されたが、CUR 2.0 静的スキーマの一部ではない製品列は、このマップ列に表示されます。 この列のキーは、ドット演算子を使用して個別の列としてクエリできます。詳細については、「データクエリ」を参照してください。	
product_comment	製品に関するコメント。	string
product_fee_code	料金を参照するコード。	string
product_fee_description	製品料金の説明。	string
product_from_location	製品の使用が開始された場所を示します。	string
product_from_location_type	製品の使用が開始された場所のタイプを示します。	string

列名	説明	データ型
product_from_region_code	AWS サービスのソースリージョンコードについて説明します。	string
product_instanceSKU	製品インスタンス SKU。	string
product_instance_family	Amazon EC2 インスタンスファミリーを示します。Amazon EC2 には、10 種類の異なるインスタンスタイプに多数のオプションが用意されています。各タイプに 1 つ以上のサイズオプションがあり、さまざまなタイプのアプリケーションのために最適化された異なるインスタンスファミリーに分類されています。	string
product_instance_type	インスタンスのタイプ、サイズ、およびファミリーを示します。これらは、インスタンスの CPU、ネットワーク、およびストレージ容量を定義します。	string
product_location	Amazon S3 バケットが存在するリージョンを示します。	string
product_location_type	タスクのエンドポイントを示します。	string

列名	説明	データ型
product_operation	この明細項目がカバーする特定の AWS オペレーションについて説明します。	string
product_pricing_unit	AWS サービスの最小請求単位。例えば、API コールあたり 0.01c です。	string
product_product_family	製品タイプのカテゴリ。	string
product_region_code	リージョンとは、データセンターがクラスター化されている世界中の物理的な場所です。 AWS では論理データセンターの各グループをアベイラビリティーゾーン (AZ) と呼びます。各 AWS リージョンは、地理的エリア内の複数の分離された物理的に分離AZsで構成されます。リージョンコード属性の名前は AWS リージョンと同じで、 AWS サービスが利用可能な場所を指定します。	string

列名	説明	データ型
product_sku	製品の一意のコード。SKU は、ProductCode 、UsageType 、および Operation を組み合わせて作成されます。サイズ柔軟な RI の場合、SKU は使用されたインスタンスを使用します。たとえば、t2.microインスタンスを使用し、使用量に RI t2.small 割引 AWS を適用した場合、明細項目 SKU はで作成されますt2.micro。	string
product_servicecode	これにより、顧客への特定の AWS サービスが一意の短い略語として識別されます。	string
product_to _location_type	サービス使用先の場所のタイプを示します。	string
product_to_location	使用先の場所を示します。	string
product_to_region_code	AWS サービスのソースリージョンコードについて説明します。	string
product_usagetype	明細項目の使用状況の詳細を表します。	string

予約列

予約列には、明細項目に適用される予約に関するデータが含まれます。

列名	説明	データ型
reservation_amortized_upfront_cost_for_usage	全額前払い RI および一部前払い RI の最初の前払い料金は、使用時間に応じて償却されます。この値は `RIAmortizedUpfrontFeeForBillingPeriod * The normalized usage amount for DiscountedUsage line items / The normalized usage amount for the RIFee` と等しくなります。前払いなし RI には前払いはないため、前払いなし RI の値は `0` です。現時点では、専有ホスト予約にはこの値を提供していません。将来のアップデートで変更を行う予定です。	double
reservation_amortized_upfront_fee_for_billing_period	この予約の前払い料金のうち、請求期間中の費用がどれくらいになるかを示します。全額前払い RI および一部前払い RI の最初の前払い料金は、今月償却されます。前払いなし RI には前払い料金はないため、前払いなし RI の値は `0` です。現時点では、専有ホスト予約にはこの値を提供してい	double

列名	説明	データ型
	ません。将来のアップデートで変更を行う予定です。	
reservation_availability_zone	この明細項目に関連付けられたリソースのアベイラビリティーゾーン。	string
reservation_effective_cost	実質的時間料金に平均化された、RI の前払い料金と時間料金の両方の合計。`EffectiveCost` は、`amortizedUpfrontCostForUsage` を `recurringFeeForUsage` に加算することにより計算されます。	double
reservation_end_time	関連付けられた RI リース期間の終了日。	string

列名	説明	データ型
reservation_modification_status	RI リースが変更されたかどうかを示します。 Original (オリジナル): 購入した RI は 1 回も変更されていません。 System (システム): 購入した RI は、コンソールまたは API を使用して変更されました。 手動： 購入した RI は AWS サポートの支援を使用して変更されました。 ManualWithData: 購入した RI は、 AWS サポートの支援と AWS RI の計算された見積りを使用して変更されました。	string
reservation_net_amortized_upfront_cost_for_usage	全前払い RI および一部前払い RI の最初の前払い料金は、該当する場合、使用時間に応じて償却されます。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
reservation_net_amortized_upfront_fee_for_billing_period	請求期間中の予約の前払い料金の費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double

列名	説明	データ型
reservation_net_effective_cost	RI の前払い料金と時間単価の合計を、平均して実効時間単価とします。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
reservation_net_recurring_fee_for_usage	継続利用料金のディスカウント後の金額。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
reservation_net_unused_amortized_upfront_fee_for_billing_period	請求期間中に償却された正味の未使用の前払い料金。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
reservation_net_unused_recurring_fee	一部前払い RI と前払いなし RI の未使用の予約時間に関連する定期的な料金。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double

列名	説明	データ型
reservation_net_upfront_value	ディスカウントが適用された RI の前払い額。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
reservation_normalized_units_per_reservation	予約サブスクリプションの各インスタンスの正規化された単位の数。	string
reservation_number_of_reservations	このサブスクリプションに含まれる予約の数。たとえば、1 件のサブスクリプションには 4 件の RI 予約を関連付けることができます。	string

列名	説明	データ型
reservation_recurring_fee_for_usage	一部前払い RI および前払いなし RI の場合、使用時間に応じて償却される定期的料金。この値は The unblended cost of the RIFee * The sum of the normalized usage amount of Usage line items / The normalized usage amount of the RIFee for size flexible Reserved Instances と等しくなります。全額前払い RI には 0 より大きい定期料金支払いがないため、全額前払い RI の値は 0 です。	double
reservation_reservation_a_r_n	この明細項目が活用した RI の Amazon リソースネーム (ARN)。これは「RI リース ID」とも呼ばれます。これは、この特定の AWS リザーブドインスタンスの一意の識別子です。値文字列には、AWS サービス名と RI が購入されたリージョンも含まれます。	string
reservation_start_time	関連付けられたリザーブドインスタンスの期間の開始日。	string

列名	説明	データ型
reservation_subscription_id	明細項目を関連オファーとマッピングする一意の識別子。RI ARN を AWS リザーブドインスタンスの ID として使用することをお勧めしますが、両方を使用できます。	string
reservation_total_reserved_normalized_units	予約サブスクリプションのすべてのインスタンスの予約済み正規化ユニットの総数。 `reservation/NormalizedUnitsPerReservation` に を乗算して、正規化されたユニットの合計を AWS 計算します`reservation/NumberOfReservations` 。	string
reservation_total_reserved_units	`TotalReservedUnits` により、Fee および RIFee の両方の明細項目に個別の値が入力されます。	string
reservation_units_per_reservation	`UnitsPerReservation` により、Fee および RIFee の両方の明細項目に個別の値が入力されます。	string

列名	説明	データ型
reservation_unused_amortized_upfront_fee_for_billing_period	全額前払い RI および一部前払い RI の最初の前払い料金の amortized-upfront-fee-for-billing-period-column 償却部分。前払いなし RI には前払いはないため、前払いなし RI の値は `0` です。現時点では、専有ホスト予約にはこの値を提供していません。将来のアップデートで変更を行う予定です。	double
reservation_unused_normalized_unit_quantity	この請求期間中に使用しなかった、サイズ柔軟なリージョン RI の未使用の正規化されたユニットの数。	double
reservation_unused_quantity	この請求期間中に使用しなかった RI 時間数。	double
reservation_unused_recurring_fee	一部前払い RI と前払いなし RI の未使用の予約時間に関連する定期的な料金。全額前払い RI には `0` より大きい定期料金がないため、全額前払い RI の値は `0` です。	double
reservation_upfront_value	AWS リザーブドインスタンスに対して支払われた前払い料金。前払いのない RI の場合、この値は `0` になります。	double

リソースタグ列

リソースタグ列には明細項目に適用されるリソースタグに関するデータが含まれます。

列名	説明	データ型
resource_tags	特定の明細項目のリソースタグのキーと値のペアとその値を含むマップ列。この列の値はすべて「文字列」データ型です。 リソースタグキーは、請求コンソールでコスト配分タグとして有効になっている場合にのみこの列に表示されます。有効にすると、特定のキーは、その値が特定の明細項目に適用される場合にのみマップ列に表示されます。 この列のキーは、ドット演算子を使用して個別の列としてクエリできます。詳細については、「データクエリ」を参照してください。	map <string, string>

Savings Plan 列

Savings Plan 列には明細項目に適用される Savings Plans に関するデータが含まれます。

列名	説明	データ型
savings_plan_amortized_upfront_commitment_for_billing_period	請求期間に Savings Plan サブスクリプションに必要と	double

列名	説明	データ型
	なる前払い金額。[全額前払い Savings Plan] と [一部前払い Savings Plan] の最初の前払い金は、当月に償却されます。前払いなし Savings Plan の場合、値は `0` です。	
savings_plan_end_time	Savings Plan 契約の有効期限日。	string
savings_plan_instance_type_family	指定した使用量に関連するインスタンスファミリー。	string
savings_plan_net_amortized_upfront_commitment_for_billing_period	請求期間中の Savings Plan サブスクリプションの前払い料金のコスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
savings_plan_net_recurring_commitment_for_billing_period	Savings Plan 手数料の混合されていない正味費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double

列名	説明	データ型
savings_plan_net_savings_plan_effective_cost	Savings Plans の実効費用。使用量を手数料で割ったものです。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
savings_plan_offering_type	購入した Savings Plan のタイプを示します。	string
savings_plan_payment_option	Savings Plan に使用できる支払いオプション。	string
savings_plan_purchase_term	Savings Plan の期間を示します。	string
savings_plan_recurring_commitment_for_billing_period	Savings Plan サブスクリプションの毎月の定額料金。例えば、[一部前払い Savings Plan] や [前払いなし Savings Plan] の毎月の定額料金が該当します。	double
savings_plan_region	サービスをホスト AWS する AWS リージョン (地理的エリア) 。このフィールドを使用して、特定の AWS リージョンの支出を分析できます。	string
savings_plan_savings_plan_a_r_n	Savings Plan の一意の ID。	string

列名	説明	データ型
savings_plan_savings_plan_effective_cost	各使用量の行に割り当てられている Savings Plan 月額料金 (前払いおよび定額) の比率。	double
savings_plan_savings_plan_rate	使用量に対する Savings Plan の料金。	double
savings_plan_start_time	Savings Plan 契約の開始日。	string
savings_plan_total_commitment_to_date	今日の該当時間までに償却された前払いコミットメントと定額コミットメントの合計。	double
savings_plan_used_commitment	使用している Savings Plan コミットメントの USD 総額。(SavingsPlanRate と使用量を乗じた額)	double

分割明細項目列

split_line_item ヘッダーの下の列は、分割コスト配分データ機能を有効にしている場合にデータエクスポートに表示されるフィールドです。詳細については、「分割コスト配分データについて」を参照してください。この機能は、Amazon ECS (Fargate を含む) AWS Batch、および Amazon EKS のみに制限されています。

列名	説明	データ型
split_line_item_actual_usage	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA	double

列名	説明	データ型
	指定された期間に Amazon ECS タスクまたは Kubernetes ポッドで発生した vCPU またはメモリの使用量 (lineItem/UsageType に基づく)。	
split_line_item_net_split_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA すべての割引が適用された後の Amazon ECS タスクまたは Kubernetes ポッドの実質的なコスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double
split_line_item_net_unused_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA すべての割引が適用された後の Amazon ECS タスクまたは Kubernetes ポッドの実効未使用コスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。	double

列名	説明	データ型
split_line_item_parent_resource_id	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA Amazon ECS タスクまたは Amazon EKS ポッドに関連付けられている親 EC2 インスタンスのリソース ID (lineItem/ResourceId 列で参照されています)。親リソース ID は、指定された期間の ECS タスクまたは Kubernetes ポッドのワークロードが親 EC2 インスタンスで実行されたことを意味します。これは、起動タイプが EC2 の Amazon ECS タスクまたは Kubernetes ポッドにのみ適用されます。	string
split_line_item_public_on_demand_split_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA パブリックオンデマンドインスタンスのレート (pricing/publicOnDemandRate 列で参照) に基づいて、その期間に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) のコスト。	double

列名	説明	データ型
split_line_item_public_on_demand_unused_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA パブリックオンデマンドインスタンスレートに基づいて、その期間に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) の未使用コスト。未使用コストとは、EC2 インスタンス (splitLineItem/ParentResourceId 列で参照) のリソース (CPU またはメモリ) で、指定された期間に使用されなかったコストです。	double
split_line_item_reserved_usage	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA 指定した期間に Amazon ECS タスクまたは Kubernetes ポッドで設定した vCPU またはメモリの使用量 (lineItem/UsageType に基づく)。	double

列名	説明	データ型
split_line_item_split_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA 期間中に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) のコスト。これには、EC2 インスタンス (splitLineItem/parentResourceId 列で参照) に予約または Savings Plans の前払いまたは一部前払い料金がある場合の償却コストが含まれます。	double
split_line_item_split_usage	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA 指定した期間に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリの使用量 (lineItem/UsageType に基づく)。これは SplitLineItem/ReservedUsage または SplitLineItem/ActualUsage の最大使用量として定義されます。	double

列名	説明	データ型
split_line_item_split_usage_ratio	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) と、EC2 インスタンスで使用可能な CPU またはメモリ全体の比率 (splitLineItem/ParentResourceId 列で参照)。	double
split_line_item_unused_cost	テーブル設定: 追加理由: INCLUDE SPLIT COST ALLOCATION DATA 期間中に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) の未使用コスト。未使用コストとは、EC2 インスタンス (splitLineItem/ParentResourceId 列で参照) のリソース (CPU またはメモリ) で、指定された期間に使用されなかったコストです。これには、EC2 インスタンス (splitLineItem/parentResourceId) に予約または Savings Plans の前払いまたは一部前払い料金がある場合の償却コストが含まれます。	double

コスト最適化の推奨事項 (Cost Optimization Hub から)

コスト最適化の推奨事項テーブルには、Cost Optimization Hub からのコスト最適化の推奨事項が含まれています。Cost Optimization Hub のレコメンデーションは AWS Compute Optimizer から統合されており、リソースの適正化、アイドル状態のリソース削除、Savings Plans、リザーブドインスタンスなど、15 種類以上の最適化で構成されています。詳細については、「AWS コスト管理ユーザーガイド」の「Cost Optimization Hub」を参照してください。

コスト最適化の推奨事項の SQL テーブル名は `COST_OPTIMIZATION_RECOMMENDATIONS` です。

テーブル設定

テーブル設定はユーザーが制御するプロパティで、ユーザーはデータエクスポートでクエリされる前にテーブルのデータまたはスキーマを変更するように設定できます。テーブル設定は JSON ステートメントとして保存され、 AWS SDK/CLI のユーザー入力またはコンソールのユーザー選択によって指定されます。

コスト最適化の推奨事項には、次のテーブル設定があります。

設定名	説明	有効値
INCLUDE_ALL_RECOMMENDATIONS	「FALSE」に設定すると、相互に互換性のない推奨事項のセットから、節約額の最も高い推奨事項のみがテーブルに保持されます。例えば、インスタンスを終了するという推奨事項と、同じインスタンスを適切なサイズにする推奨事項では、「インスタンスを終了する」のみが保持されます。 「TRUE」に設定すると、すべての推奨事項がテーブルに保持されます。	TRUE、FALSE

設定名	説明	有効値
	これは、Cost Optimization Hub コンソールでグループ関連の推奨事項とも呼ばれます。詳細については、「AWS コスト管理ユーザーガイド」の「関連する推奨事項のグループ化」を参照してください。	

設定名	説明	有効値
FILTER	これにより、さまざまな推奨事項属性に基づいて推奨事項をフィルタリングできます。節約重複排除アルゴリズムが適用される前に、フィルターがテーブルに適用されます。 Cost Optimization Hub コンソールと同じパラメータを使用してフィルタリングできます。詳細については、「AWS コスト管理ユーザーガイド」の「コスト最適化の機会の優先順位付け」を参照してください。 この構成には、Cost Optimization Hub `list-recommendations` API の `filter` パラメータで使用されるものと同じ JSON 構造を使用してフィルターステートメントが指定されます。これは、JSON 文字列として指定する必要があります。構造の詳細については、「`list-recommendations` 構造」を参照してください。	Cost Optimization Hub `list-recommendations` API に有効な JSON 文字列。

サービスリンクロール

コスト最適化の推奨事項テーブルのエクスポートを作成するには、データエクスポートのサービスにリンクされたロールが必要です。サービスにリンクされたロールを作成する方法については、

「AWS コスト管理ユーザーガイドの「データエクスポートのサービスにリンクされたロール」を参照してください。

AWS Organizations のサポート

Cost Optimization Hub は AWS Organizations と統合して、管理アカウントが Cost Optimization Hub でメンバーアカウントのレコメンデーションを表示できるかどうかを制御します。詳細については、「AWS コスト管理ユーザーガイド」の「Cost Optimization Hub の開始方法」を参照してください。

特定の AWS アカウントのコスト最適化レコメンデーションテーブルは、Cost Optimization Hub 用に設定したのと同じ AWS Organizations 設定を継承します。つまり、コスト最適化レコメンデーションテーブルの内容は、各 AWS アカウントの Cost Optimization Hub に表示されるレコメンデーションと一致します。

コスト最適化の推奨事項列

列名	説明	データ型	Null 値を使用できます
account_ID	推奨事項の対象となるアカウント。	string	いいえ
action_type	推奨事項を採用することによって実行できるアクションのタイプ。	string	いいえ
currency_code	推奨事項に使用される通貨コード。	string	いいえ
current_resource_details	JSON 文字列形式のリソースの詳細。	string	はい
current_resource_summary	現在のリソースの説明。	string	はい
current_resource_type	リソースのタイプ。	string	はい
estimated_monthly_cost_after_discount	割引後の現在のリソースの推定月額コス	double	はい

列名	説明	データ型	Null 値を使用できます
	ト。リザーブドインスタンスと Savings Plans の場合、対象となる使用量のコストを指します。		
estimated_monthly_cost_before_discount	割引前の現在のリソースの推定月額コスト。リザーブドインスタンスと Savings Plans の場合、対象となる使用量のコストを指します。	double	いいえ
estimated_monthly_savings_after_discount	割引後の推奨事項の推定月額節約額。	double	はい
estimated_monthly_savings_before_discount	割引前の推奨事項の推定月額節約額。	double	いいえ
estimated_savings_percentage_after_discount	コスト計算のルックバック期間の合計コストに対する割引後の推定節約率。	double	はい
estimated_savings_percentage_before_discount	コスト計算のルックバック期間の合計コストに対する割引前の推定節約率。	double	いいえ
implementation_effort	推奨事項を実装するために必要な労力。	string	いいえ

列名	説明	データ型	Null 値を使用できます
last_refresh_timestamp	推奨事項が最後に生成された時刻。	timestamp	いいえ
recommendation_ID	推奨事項の ID。	string	いいえ
recommendation_lookback_period_in_days	推奨事項の生成に使用されるルックバック期間。	integer	いいえ
recommendation_source	推奨事項のソース。	string	いいえ
recommended_resource_details	JSON 文字列形式の推奨リソースに関する詳細。	string	はい
recommended_resource_summary	推奨リソースの説明。	string	はい
recommended_resource_type	推奨事項のリソースタイプ。	string	はい
region	リソースの AWS リージョン。	string	はい
resource_arn	リソースの Amazon リソースネーム (ARN)。	string	はい
restart_needed	推奨事項の実装に再起動が必要かどうか。	boolean	いいえ
rollback_possible	推奨事項の実装をロールバックできるかどうか。	boolean	いいえ

列名	説明	データ型	Null 値を使用できます
tags	推奨事項が存在するリソースに関連付けられたタグのリスト。	map	はい

AWS 列を含む FOCUS 1.0

AWS 列を含む FOCUS 1.0 テーブルには、FinOps Open Cost and Usage Specification (FOCUS) 1.0 でフォーマットされたコストと使用状況のデータと、独自の請求データを含む AWS からの 5 つの追加列が含まれています。これらの列は、x_CostCategories、x_Discounts、x_Operation、x_ServiceCode、x_UsageType です。FOCUS オープンソース仕様の詳細については、FOCUS ウェブサイトを参照してください。

AWS 列を含む FOCUS 1.0 の SQL テーブル名は です。 `FOCUS_1_0_AWS`

テーブル設定

AWS 列テーブルを持つ FOCUS 1.0 のテーブル設定はありません。

AWS Organizations のサポート

AWS 列を含む FOCUS 1.0 テーブルは、 AWS Organizations の統合請求機能で行った設定を継承します。一括決済が有効になっている場合、管理アカウントとメンバーアカウントにはさまざまな動作があります。管理アカウントを使用している場合、 AWS 列テーブルを含む FOCUS 1.0 には、管理アカウントと組織内のすべてのメンバーアカウントのコストと使用状況データが含まれます。メンバーアカウントを使用している場合、 AWS 列テーブルを含む FOCUS 1.0 には、そのメンバーアカウントのコストと使用状況データのみが含まれます。

組織に加入すると、メンバーアカウントは、そのアカウントが組織のメンバーであった期間のデータのみをエクスポートできます。例えば、あるメンバーアカウントが組織 A を離れ、同月 15 日に組織 B に加入したとします。その後、メンバーアカウントはエクスポートを作成します。メンバーアカウントは組織 B に参加した後にエクスポートを作成しているため、メンバーアカウントの AWS FOCUS 1.0 のエクスポートには、アカウントが組織 B のメンバーであった期間のコストと使用状況データのみが含まれます。

AWS 列を含む FOCUS 1.0

列名	説明	データ型	Null 値を使用できます
AvailabilityZone	高可用性と耐障害性を提供するリージョン内の物理的に分離されたエリアと分離されたエリアのプロバイダー割り当て識別子。	string	はい
BilledCost	請求書のベースとなる料金。すべての割引料金と割引を含みますが、将来の対象料金をカバーするために支払われた関連購入の償却は除きます。	double	いいえ
BillingAccountId	請求アカウントのプロバイダー割り当て識別子。	string	いいえ
BillingAccountName	請求アカウントのプロバイダー割り当て名。	string	はい
BillingCurrency	リソースまたはサービスの料金が請求された通貨を表す識別子。	string	いいえ
BillingPeriodEnd	請求期間の終了日時。	timestamp_milliseconds	いいえ

列名	説明	データ型	Null 値を使用できます
BillingPeriodStart	請求期間の開始日時。	timestamp_milliseconds	いいえ
ChargeCategory	行が、前払い料金または定期料金、既に発生した使用コスト、事後調整 (クレジットなど)、または税金を表しているかどうかを示すインジケータ。	string	いいえ
ChargeClass	行が、通常の料金を表すか、1 つ以上の以前の料金に対する修正を表すかを示すインジケータ。	string	はい
ChargeDescription	追加の検出を必要としない行のおおまかなコンテキスト。	string	はい
ChargeFrequency	料金が発生する頻度のインジケータ。	string	いいえ
ChargePeriodEnd	請求期間の終了日時。	timestamp_milliseconds	いいえ
ChargePeriodStart	請求期間の開始日時。	timestamp_milliseconds	いいえ

列名	説明	データ型	Null 値を使用できます
CommitmentDiscountCategory	CommitmentDiscountId 列で識別されるコミットメントベースの割引が、使用量またはコスト (つまり支出) に基づいているかどうかを示すインジケータ。	string	はい
CommitmentDiscountId	コミットメントベースの割引のプロバイダー割り当て識別子。	string	はい
CommitmentDiscountName	コミットメントベースの割引に割り当てられた表示名。	string	はい
CommitmentDiscountStatus	料金が使用済みまたは未使用のコミットメント割引に対応するかどうかを示すインジケータ。	string	はい
CommitmentDiscountType	行に適用されるコミットメントベースの割引のタイプを識別するプロバイダー割り当ての名前。	string	はい
ConsumedQuantity	使用または購入した特定のリソースまたはサービスの ConsumedUnit に基づくボリューム。	double	はい

列名	説明	データ型	Null 値を使用できます
ConsumedUnit	プロバイダーがリソースまたはサービスに関連付けられた特定の SKU の使用状況を測定する方法を示すプロバイダー割り当ての測定単位。	string	はい
ContractedCost	ContractedUnitPrice と対応する PricingQuantity を掛けて計算されたコスト。	double	いいえ
ContractedUnitPrice	関連付けられた SKU の単一の PricingUnit の合意された単価。交渉された割引は含まれるが、交渉されたコミットメントベースの割引やその他の割引は除外されます。	double	はい
EffectiveCost	すべての割引料金と割引が含まれたコスト。将来の対象料金をカバーするために支払われる関連購入 (1 回限りまたは定期) の償却が加算されます。	double	いいえ

列名	説明	データ型	Null 値を使用できます
InvoiceIssuerName	消費されたソースまたはサービスの請求を担当するエンティティ。一般的にコスト分析やレポートシナリオに使用されます。	string	いいえ
ListCost	ListUnitPrice と対応する PricingQuantity を掛けて計算されたコスト。	double	いいえ
ListUnitPrice	プロバイダーによって公開された、関連付けられた SKU の単一の PricingUnit に対する推奨単価 (割引は除く)。	double	はい
PricingCategory	使用時または購入時の料金に使用される料金モデル。	string	はい
PricingQuantity	PricingUnit に基づいて、使用または購入したリソースまたはサービスに関連付けられた特定の SKU のボリューム。	double	はい

列名	説明	データ型	Null 値を使用できます
PricingUnit	単価を決定するためのプロバイダー割り当ての測定単位。ブロック料金などの料金ルールを適用した後、プロバイダーの料金が使用量と購入数量をどのように測定したかを示します。	string	はい
ProviderName	リソースまたはサービスを購入可能にしたエンティティ。	string	いいえ
PublisherName	購入したリソースまたはサービスを作成したエンティティ。	string	いいえ
RegionId	リソースがプロビジョニングされる、またはサービスが提供される独立した地理的エリアに対してプロバイダーが割り当てた識別子。	string	はい
RegionName	リソースがプロビジョニングされる、またはサービスが提供される独立した地理的エリアの名前。	string	はい

列名	説明	データ型	Null 値を使用できます
ResourceId	リソースのプロバイダー割り当て識別子。	string	はい
ResourceName	リソースに割り当てられた表示名。	string	はい
ResourceType	料金が適用されるリソースのタイプ。	string	はい
ServiceCategory	サービスのコア関数に基づくサービスのおおまかな分類。	string	いいえ
ServiceName	購入したオファリングの表示名。	string	いいえ
SkuId	1 つ以上の SKU 料金に共通するプロパティを整理するための、プロバイダーがサポートする構造を定義する一意の識別子。	string	はい
SkuPriceId	料金の計算に使用される単価を定義する一意の識別子。	string	はい
SubAccountId	リソースまたはサービスのグループに割り当てられた ID。多くの場合、アクセスやコストの管理に使用されます。	string	はい

列名	説明	データ型	Null 値を使用できます
SubAccountName	リソースまたはサービスのグループに割り当てられた名前。多くの場合、アクセスやコストの管理に使用されます。	string	はい
Tags	タグソースに割り当てられたタグのセット。プロバイダー定義またはユーザー定義のタグ評価の可能性も考慮されます。	map <string, string>	はい
x_CostCategories	特定の明細項目のコストカテゴリのキーと値のペアとその値を含むマップ列。	map <string, string>	はい
x_Discounts	この明細項目に適用される特定の割引のキーと値のペアを含むマップ列。	map <string, double>	はい
x_Operation	この明細項目の対象となる特定の AWS オペレーション。これは、明細項目の特定の使用状況を表します。	string	はい
x_ServiceCode	この明細項目で使用されるサービスのコード。	string	はい

列名	説明	データ型	Null 値を使用できます
x_UsageType	この明細項目の使用状況の詳細。	string	はい

列の適合ギャップを含む AWS FOCUS 1.0

次の表は、 AWS 列テーブルを含む FOCUS 1.0 のエクスポートに存在する可能性のあるすべての適合ギャップを示しています。関連するシナリオのコストと使用状況データを受信していない場合、特定の適合ギャップはエクスポートに適用されません。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
欠落データ	ContractedUnitPrice	ContractedUnitPrice は、修正されていない使用状況と購入の ChargeCategory 明細項目に対して null にすることはできません。	ContractedUnitPrice は、一部の製品では null である場合があります。
	InvoiceIssuerName	InvoiceIssuerName を null にすることはできません。	一部の料金では、InvoiceIssuerName が null である場合があります。
	ListUnitPrice	修正されていない使用量と購入の ChargeCategory 明細項目では、ListUnitPrice を null にすることはできません。	ListUnitPrice は、一部の製品では null である場合があります。
	PricingUnit	修正ではない Usage および Purchase	一部の製品では PricingUnit が null で

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
		ChargeCategory の明細項目の場合、PricingUnit は null にできません。	ある場合があります。
	PublisherName	PublisherName を null にすることはできません。	PublisherName は、特定の料金で null になる場合があります。
	SkuId	SkuId は、修正されていない使用状況と購入の ChargeCategory 明細項目に対して null にすることはできません。	SkuId は、一部の製品では null である場合があります。
	SkuPriceId	修正されていない使用状況と購入の ChargeCategory 明細項目では、SkuPriceId を null にすることはできません。	一部の明細項目の SkuPriceId で、null になるべきではないのに、null になっている場合があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
誤ったデータ	ConsumedQuantity	ConsumedQuantity は、実際に使用した使用量を示すために必要な列です。	ConsumedQuantity には、請求された使用量が含まれます。つまり、特定のサービスに最小請求数量が適用されている場合、ConsumedQuantity が正しくなくなる可能性があります。 例えば、Athena クエリには 10MB 以上、Glue クローラーには 10 分の最小実行数があります。これらのサービスの場合、ConsumedQuantity は最小請求量を含む値を表示します。

AWS 列を含む FOCUS 1.0 (プレビュー)

> **Note**
>
> AWS 列を含む FOCUS 1.0 が、新しいデータエクスポートテーブルの一般提供 (GA) でリリースされました。このページは、 AWS 列を含む FOCUS 1.0 のプレビューリリース用です。FOCUS 1.0 の GA テーブルを AWS 列とともに使用します。

AWS 列を含む FOCUS 1.0 (プレビュー) テーブルには、FinOps Open Cost and Usage Specification (FOCUS) 1.0 でフォーマットされたコストと使用状況のデータと、

独自の請求データを含む AWS からの 5 つの追加列が含まれています。これらの列は、x_CostCategories、x_Discounts、x_Operation、x_ServiceCode、x_UsageType です。FOCUS オープンソース仕様の詳細については、FOCUS ウェブサイトを参照してください。

AWS 列を含む FOCUS 1.0 の SQL テーブル名 (プレビュー) は です。 `FOCUS_1_0_AWS_PREVIEW`

プレビューメモ

AWS 列を含む FOCUS 1.0 (プレビュー) テーブルは間もなく廃止されます。

すべてのお客様は、 AWS 列で FOCUS 1.0 の GA リリースを使用することをお勧めします。現在、AWS 列 (プレビュー) テーブルで FOCUS 1.0 を使用している場合は、できるだけ早く GA テーブルに切り替える必要があります。GA テーブルは仕様適合ギャップの数を大幅に削減し、本番稼働用 FinOps プロセスに適しています。GA テーブルの適合ギャップのリストについては、「FOCUS 1.0 with AWS columns conformance gap」を参照してください。

本番ワークロードでは、 AWS 列 (プレビュー) を含む FOCUS 1.0 はお勧めしません。プレビューテーブルの適合ギャップのリストについては、AWS 「列を含む FOCUS 1.0 (プレビュー) 適合ギャップ」を参照してください。

テーブル設定

AWS 列 (プレビュー) テーブルを持つ FOCUS 1.0 のテーブル設定はありません。

AWS Organizations のサポート

AWS 列を含む FOCUS 1.0 (プレビュー) テーブルは、 AWS Organizations の一括請求機能で行った設定を継承します。一括決済が有効になっている場合、管理アカウントとメンバーアカウントにはさまざまな動作があります。管理アカウントを使用している場合、 AWS 列を含む FOCUS 1.0 (プレビュー) テーブルには、管理アカウントと組織内のすべてのメンバーアカウントのコストと使用状況データが含まれます。メンバーアカウントを使用している場合、 AWS 列を含む FOCUS 1.0 (プレビュー) テーブルには、そのメンバーアカウントのコストと使用状況データのみが含まれます。

組織に加入すると、メンバーアカウントは、そのアカウントが組織のメンバーであった期間のデータのみをエクスポートできます。例えば、あるメンバーアカウントが組織 A を離れ、同月 15 日に組織 B に加入したとします。その後、メンバーアカウントはエクスポートを作成します。メンバーアカウントは組織 B に加わった後にエクスポートを作成しているため、メンバーアカウントの FOCUS 1.0 のその月の AWS 列 (プレビュー) を含むエクスポートには、アカウントが組織 B のメンバーであった期間のコストと使用状況データのみが含まれます。

AWS 列を含む FOCUS 1.0 (プレビュー) の列

> **Note**
>
> AWS 列を含む FOCUS 1.0 が、新しいデータエクスポートテーブルの一般提供 (GA) でリリースされました。このページは、 AWS 列を含む FOCUS 1.0 のプレビューリリース用です。FOCUS 1.0 の GA テーブルを AWS 列とともに使用します。

列名	説明	データ型	Null 値を使用できます
AvailabilityZone	高可用性と耐障害性を提供するリージョン内の物理的に分離されたエリアと分離されたエリアのプロバイダー割り当て識別子。	string	はい
BilledCost	請求書のベースとなる料金。すべての割引料金と割引を含みますが、将来の対象料金をカバーするために支払われた関連購入の償却は除きます。	double	いいえ
BillingAccountId	請求アカウントのプロバイダー割り当て識別子。	string	いいえ
BillingAccountName	請求アカウントのプロバイダー割り当て名。	string	はい

列名	説明	データ型	Null 値を使用できます
BillingCurrency	リソースまたはサービスの料金が請求された通貨を表す識別子。	string	いいえ
BillingPeriodEnd	請求期間の終了日時。	timestamp	いいえ
BillingPeriodStart	請求期間の開始日時。	timestamp	いいえ
ChargeCategory	行が、前払い料金または定期料金、既に発生した使用コスト、事後調整 (クレジットなど)、または税金を表しているかどうかを示すインジケータ。	string	いいえ
ChargeClass	行が、通常の料金を表すか、1 つ以上の以前の料金に対する修正を表すかを示すインジケータ。	string	はい
ChargeDescription	追加の検出を必要としない行のおおまかなコンテキスト。	string	はい
ChargeFrequency	料金が発生する頻度のインジケータ。	string	いいえ
ChargePeriodEnd	請求期間の終了日時。	timestamp	いいえ

列名	説明	データ型	Null 値を使用できます
ChargePeriodStart	請求期間の開始日時。	timestamp	いいえ
CommitmentDiscountCategory	CommitmentDiscountId 列で識別されるコミットメントベースの割引が、使用量またはコスト (つまり支出) に基づいているかどうかを示すインジケータ。	string	はい
CommitmentDiscountId	コミットメントベースの割引のプロバイダー割り当て識別子。	string	はい
CommitmentDiscountName	コミットメントベースの割引に割り当てられた表示名。	string	はい
CommitmentDiscountStatus	料金が使用済みまたは未使用のコミットメント割引に対応するかどうかを示すインジケータ。	string	はい
CommitmentDiscountType	行に適用されるコミットメントベースの割引のタイプを識別するプロバイダー割り当ての名前。	string	はい

列名	説明	データ型	Null 値を使用できます
ConsumedQuantity	使用または購入した特定のリソースまたはサービスの ConsumedUnit に基づくボリューム。	double	はい
ConsumedUnit	プロバイダーがリソースまたはサービスに関連付けられた特定の SKU の使用状況を測定する方法を示すプロバイダー割り当ての測定単位。	string	はい
ContractedCost	ContractedUnitPrice と対応する PricingQuantity を掛けて計算されたコスト。	double	いいえ
ContractedUnitPrice	関連付けられた SKU の単一の PricingUnit の合意された単価。交渉された割引は含まれるが、交渉されたコミットメントベースの割引やその他の割引は除外されます。	double	はい

列名	説明	データ型	Null 値を使用できます
EffectiveCost	すべての割引料金と割引が含まれたコスト。将来の対象料金をカバーするために支払われる関連購入 (1 回限りまたは定期) の償却が加算されます。	double	いいえ
InvoiceIssuerName	消費されたソースまたはサービスの請求を担当するエンティティ。一般的にコスト分析やレポートシナリオに使用されます。	string	いいえ
ListCost	ListUnitPrice と対応する PricingQuantity を掛けて計算されたコスト。	double	いいえ
ListUnitPrice	プロバイダーによって公開された、関連付けられた SKU の単一の PricingUnit に対する推奨単価 (割引は除く)。	double	はい
PricingCategory	使用時または購入時の料金に使用される料金モデル。	string	はい

列名	説明	データ型	Null 値を使用できます
PricingQuantity	PricingUnit に基づいて、使用または購入したリソースまたはサービスに関連付けられた特定の SKU のボリューム。	double	はい
PricingUnit	単価を決定するためのプロバイダー割り当ての測定単位。ブロック料金などの料金ルールを適用した後、プロバイダーの料金が使用量と購入数量をどのように測定したかを示します。	string	はい
ProviderName	リソースまたはサービスを購入可能にしたエンティティ。	string	いいえ
PublisherName	購入したリソースまたはサービスを作成したエンティティ。	string	いいえ
RegionId	リソースがプロビジョニングされる、またはサービスが提供される独立した地理的エリアに対してプロバイダーが割り当てた識別子。	string	はい

列名	説明	データ型	Null 値を使用できます
RegionName	リソースがプロビジョニングされる、またはサービスが提供される独立した地理的エリアの名前。	string	はい
ResourceId	リソースのプロバイダー割り当て識別子。	string	はい
ResourceName	リソースに割り当てられた表示名。	string	はい
ResourceType	料金が適用されるリソースのタイプ。	string	はい
ServiceCategory	サービスのコア関数に基づくサービスのおおまかな分類。	string	いいえ
ServiceName	購入したオファリングの表示名。	string	いいえ
SkuId	1 つ以上の SKU 料金に共通するプロパティを整理するための、プロバイダーがサポートする構造を定義する一意の識別子。	string	はい
SkuPriceId	料金の計算に使用される単価を定義する一意の識別子。	string	はい

列名	説明	データ型	Null 値を使用できます
SubAccountId	リソースまたはサービスのグループに割り当てられた ID。多くの場合、アクセスやコストの管理に使用されます。	string	はい
SubAccountName	リソースまたはサービスのグループに割り当てられた名前。多くの場合、アクセスやコストの管理に使用されます。	string	はい
Tags	タグソースに割り当てられたタグのセット。プロバイダー定義またはユーザー定義のタグ評価の可能性も考慮されます。	map	はい
x_CostCategories	特定の明細項目のコストカテゴリのキーと値のペアとその値を含むマップ列。	map	はい
x_Discounts	この明細項目に適用される特定の割引のキーと値のペアを含むマップ列。	map	はい

列名	説明	データ型	Null 値を使用できます
x_Operation	この明細項目の対象となる特定の AWS オペレーション。これは、明細項目の特定の使用状況を表します。	string	いいえ
x_ServiceCode	この明細項目で使用されるサービスのコード。	string	いいえ
x_UsageType	この明細項目の使用状況の詳細。	string	いいえ

AWS 列を含む FOCUS 1.0 (プレビュー) の適合ギャップ

> **Note**
>
> AWS 列を含む FOCUS 1.0 が、新しいデータエクスポートテーブルの一般提供 (GA) でリリースされました。このページは、 AWS 列を含む FOCUS 1.0 のプレビューリリース用です。FOCUS 1.0 の GA テーブルを AWS 列とともに使用します。

次の表は、 AWS 列を含む FOCUS 1.0 (プレビュー) テーブルのエクスポートに存在する可能性のあるすべての適合ギャップを示しています。関連するシナリオのコストと使用状況データを受信していない場合、特定の適合ギャップはエクスポートに適用されません。例えば、エクスポートに以前の請求書の修正である明細項目が含まれていない場合、ChargeClass 列の適合ギャップは適用されません。

この時点で、個々の適合ギャップがコストと使用状況の分析に関連しているかどうかを特定する必要があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
欠落データ	ChargeClass	ChargeClass は、特定の明細項目が過去の請求期間の修正であるかどうかを示すために必要な列です。	現時点では、この列の修正として識別できる明細項目はありません。
	ContractedUnitPrice	ContractedUnitPrice は、修正されていない使用状況と購入の ChargeCategory 明細項目に対して null にすることはできません。	ContractedUnitPrice は、一部の製品では null である場合があります。
	InvoiceIssuerName	InvoiceIssuerName を null にすることはできません。	一部の料金では、InvoiceIssuerName が null である場合があります。
	ListUnitPrice	修正されていない使用量と購入の ChargeCategory 明細項目では、ListUnitPrice を null にすることはできません。	ListUnitPrice は、一部の製品では null である場合があります。
	PricingUnit	修正ではない Usage および Purchase ChargeCategory の明細項目の場合、PricingUnit は null にできません。	一部の製品では PricingUnit が null である場合があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
	ProviderName	ProviderName を null にすることはできません。	ProviderName は、一部の料金で null である場合があります。
	PublisherName	PublisherName を null にすることはできません。	PublisherName は、特定の料金で null になる場合があります。
	ResourceName	ResourceName は、ユーザーが特定のリソースに割り当てた名前を表示するために必要な列です。	この列には、EC2 インスタンスと EBS ボリュームリソース名のみがカスタマーによって割り当てられます。この列には他のリソース名は表示されません。タグとして割り当てられたリソース名は、タグ列に表示されます。
	SkuId	SkuId は、修正されていない使用状況と購入の ChargeCategory 明細項目に対して null にすることはできません。	SkuId は、一部の製品では null である場合があります。
	SkuPriceId	修正されていない使用状況と購入の ChargeCategory 明細項目では、SkuPriceId を null にすることはできません。	一部の明細項目の SkuPriceId で、null になるべきではないのに、null になっている場合があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
誤ったデータ	BilledCost	BilledCost の合計は、請求書と一致する必要があります。	BilledCost の合計が請求書と一致しない場合があります。
	ConsumedQuantity	ConsumedQuantity は、実際に使用した使用量を示すために必要な列です。	ConsumedQuantity には、請求された使用量が含まれます。つまり、特定のサービスに最小請求数量が適用されている場合、ConsumedQuantity が正しくなくなる可能性があります。 例えば、Athena クエリには 10MB 以上、Glue クローラーには 10 分の最小実行数があります。これらのサービスの場合、ConsumedQuantity は最小請求量を含む値を表示します。
	ContractedCost	ContractedCost は、修正されていない料金の ContractedUnitPrice x PricingQuantity と等しくなければなりません。	ContractedCost は、クレジットなどの特定の料金について ListUnitPrice や PricingQuantity と一致しない場合があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
	ContractedCost	ContractedCost には、プロバイダーと合意した料金が反映されている必要があります。	Savings Plans またはリザーブドインスタンスの対象となる使用状況明細項目の場合、ContractedCost は割引が適用された場合でも ListCost に等しくなります。
	ContractedUnitPrice	ContractedUnitPrice には、プロバイダーと合意した料金が反映されている必要があります。	Savings Plans またはリザーブドインスタンスの対象となる使用状況明細項目の場合、ContractedUnitPrice は割引が適用されていても ListUnitPrice に等しくなります。
	ListCost	ListCost は、修正されていない料金の ListUnitPrice x PricingQuantity と等しくなければなりません。	ListCost は、クレジットなどの特定の料金について ListUnitPrice および PricingQuantity と一致しない場合があります。
	ListCost	ListCost は ChargeCategory of Credit の BilledCost と等しくなければなりません。	ListCost は ChargeCategory of Credit の BilledCost と等しくない場合があります。

適合ギャップのタイプ	影響を受ける列	FOCUS 1.0 の要件	適合ギャップの説明
	PricingCategory	PricingCategory は、すべてのスポット料金に対して「Dynamic」である必要があります。	特定の EC2 スポット明細項目では、PricingCategory が正しくない場合があります。
	Tags	タグは、特定の明細項目に適用されるタグを表示するために必要な列です。 この列では、プロバイダーが (i) プロバイダーが作成したタグのキーにのみプレフィックスを追加し、(ii) タグ名を変更しないことが求められます。	2 つのギャップがあります。 (i) ユーザー作成のタグキーには、ユーザーが作成したタグであることを示すプレフィックスがあります。 (ii) プロバイダーとユーザーのタグキー名は、大文字、スペース、「_」以外の特殊文字を持たないように正規化されます。正規化によってタグキーの競合が発生する場合、保持されるタグ値は 1 つだけです。

コストと使用状況ダッシュボード

コストと使用状況ダッシュボードの SQL テーブル名は `COST_AND_USAGE_DASHBOARD` です。

テーブル設定

テーブル設定はユーザーが制御するプロパティで、ユーザーはデータエクスポートでクエリされる前にテーブルのデータまたはスキーマを変更するように設定できます。

コストと使用状況ダッシュボードには変更可能なテーブル設定はありません。

AWS Organizations のサポート

コストと使用状況ダッシュボードテーブルは CUR 2.0 データから生成されます。つまり、CUR 2.0 に適用されるのと同じ AWS Organizations 設定を継承し、同じ動作が適用されます。 AWS Organizations がコストと使用状況ダッシュボードにどのように適用されるかを理解するには、CUR 2.0 セクションの AWS Organizations サポートを参照してください。

コストと使用状況ダッシュボード

列名	説明
amortized_cost	前払いの実質的なコストと、請求期間にわたる毎月の予約手数料を反映。これは、明細項目の種類に基づくコストの合計です。コストは次のように決定されます。 • 明細項目タイプが「SavingsPlanCovered Usage」の場合、コストは Savings Plans の実質的なコストです。 • 明細項目タイプが「SavingsPlanRecurringFee」の場合、コストはこれまでの Savings Plans の合計コミットメントから使用済みコミットメントを引いたものです。 • 明細項目タイプが「SavingsPlanNegation」または「SavingsPlanUpfrontFee」の場合、コストは 0 になります。 • 明細項目タイプが「DiscountedUsage」の場合、コストは予約の実質的なコストです。

列名	説明
	• 明細項目タイプが「RIFee」の場合、コストは、請求期間中の未使用の償却前払い料金と、予約の未使用の定期料金の合計です。 • 明細項目タイプがFee「」で、予約 ARN がある場合、コストは 0 です。 • 他のすべての明細項目タイプの場合、コストは明細項目のブレンドされていないコストになります。
availability_zone	この明細項目をホストするアベイラビリティーゾーン。例えば、us-east-1a または us-east-1b などです。
billing_entity	請求書または取引が AWS Marketplace 向けか、他の AWS サービスの購入向けかを特定するのに役立ちます。可能な値は以下のとおりです: AWS： Marketplace 以外の AWS サービスのトランザクションを識別します AWS 。 AWS Marketplace: AWS Marketplace での購入を示します。
billing_period	ダッシュボードの対象となる請求期間の開始日 (UTC)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。 例: 2023-10-01T00:00:00.000Z

列名	説明
charge_category	この明細項目の対象となる請求のカテゴリ。取り得るカテゴリには以下のものがあります。 Running_usage: charge_typeがDiscountedUsage」、SavingsPlanCoveredUsage」、または「Usage」のいずれかである場合。 Non_Usage: の他のすべて用charge_type。
charge_type	この明細項目の対象となる請求の種類。可能なすべての値については、lineItem/LineItemTypeこちらを参照してください。
current_generation	Amazon RDS インスタンスが現在の世代のものかどうかを識別するのに役立ちます。
database_engine	データベースのデータベースエンジンを示します。 例: PostgreSQL、Oracle。

列名	説明
instance_type	インスタンスのタイプ、サイズ、およびファミリーを示します。これらは、インスタンスのCPU、ネットワーク、およびストレージ容量を定義します。 例: t2.small、m4.xlarge、t2.micro、m4.large、t2.large サービス: • Amazon EC2 • Amazon RDS • OpenSearch Service • Amazon ElastiCache • Amazon EMR • 完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。
instance_type_family	指定された使用に関連付けられているインスタンスファミリー。 例: t2、m4、m3 サービス: • Amazon DocumentDB • Amazon RDS

列名	説明
invoice_id	特定の明細項目に関連付けられた ID。invoice_id は、エクスポートが確定するまで空白のままです。
item_description	明細項目タイプの説明。たとえば、使用料の明細項目は、特定の期間に発生した使用タイプを要約したものです。 サイズ柔軟な RI の場合、説明は、利点が適用された RI に対応します。たとえば、明細項目が t2.micro に対応し、t2.small RI が使用に適用された場合、 は t2.small lineItem/LineItemDescriptionを表示します。 RI ディスカウントが適用された使用料の明細項目の説明には、明細項目の対象となる料金プランが含まれています。
legal_entity	特定の製品またはサービスの登録販売者。ほとんどの場合、請求書を発行する事業体と法律上の事業体は同じです。サードパーティーの AWS Marketplace トランザクションでは、値が異なる場合があります。可能な値は以下のとおりです: Amazon Web Services, Inc. : AWS のサービスを販売する事業体。 Amazon Web Services India Private Limited: インドの AWS サービスのリセラーとして機能するインドのローカルエンティティ。

列名	説明
linked_account_id	この明細項目を使用したアカウントの ID。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。このフィールドを使用して、アカウント別のコストや使用状況を追跡することができます。
linked_account_name	この明細項目を使用したアカウントの名前。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。このフィールドを使用して、アカウント別のコストや使用状況を追跡することができます。
operation	この明細項目の対象となる特定の AWS オペレーション。これは、明細項目の特定の使用状況を表します。例えば、RunInstances の値は Amazon EC2 インスタンスのオペレーションを示します。
payer_account_id	支払いアカウントのアカウント ID。 AWS の組織の場合、これは管理アカウントのアカウント ID です。
payer_account_name	支払いアカウントのアカウント名。 AWS の組織の場合、これは管理アカウントのアカウントの名前です。

列名	説明
platform	Amazon EC2 インスタンスのオペレーティングシステムを示します。 例: Amazon Linux、Ubuntu、Windows Server、Oracle Linux、FreeBSD サービス: • Amazon AppStream • Amazon EC2 • Amazon GameLift • Amazon Lightsail • Amazon WorkSpaces • Amazon CodeBuild
pricing_unit	が使用コストの計算 AWS に使用した料金単位。例えば、Amazon EC2 インスタンスの使用料の料金単位は時間数です。

列名	説明
processor	Amazon EC2 インスタンス上のプロセッサを示します。 例: High Frequency Intel Xeon E7-8880 v3 (Haswell)、Intel Xeon E5-2670、AMD EPYC 7571 サービス: • Amazon DocumentDB • Amazon EC2 • Amazon Neptune • Amazon RDS • AWS データベース移行サービス
processor_features	インスタンスのプロセッサ機能を表します。 例: Intel AVX、Intel AVX2、Intel AVX512、Intel Turbo サービス: • AWS データベース移行サービス • Amazon DocumentDB • Amazon EC2 • Amazon Neptune • Amazon RDS

列名	説明
product_code	測定された製品のコード。例えば、AmazonEC2 は、Amazon Elastic Compute Cloud の製品コードです。
product_family	製品タイプのカテゴリ。 例： アラーム、 AWS 予算、停止したインスタンス、ストレージスナップショット、コンピューティング
product_from_location	製品の使用が開始された場所を示します。 サンプル値: 外部、米国東部 (バージニア北部)、グローバル サービス: • Amazon CloudFront • AWS DataTransfer

列名	説明
product_group	定義上類似している、またはグループ化された複数の製品の構成。たとえば、Amazon EC2 チームは製品を共有インスタンス、専用ホスト、専用用途に分類できます。 サービス: • AWS Certificate Manager • AWS CodeCommit • AWS Glue • AWS IoT Analytics • AWS Lambda
product_name	AWS サービスのフルネーム。 例： AWS Backup、 AWS Config、Amazon Registrar、Amazon Elastic File System、Amazon Elastic Compute Cloud
product_to_location	使用先の場所を示します。 サンプル値: 外部、米国東部 (バージニア北部) サービス: • Amazon CloudFront • AWS DataTransfer

列名	説明
public_cost	パブリックオンデマンドインスタンスレートに基づく明細項目の合計コスト。複数のオンデマンドパブリックコストが発生する SKU がある場合は、最も高い階層の同等のコストが表示されます。無料利用枠や階層型価格を提供するサービスなどが該当します。
purchase_option	この明細項目で AWS リソースを取得して支払いを行う方法。purchase_option 列には、Savings Plans、リザーブドインスタンス、スポットインスタンスをそれぞれ表す「SavingsPlan」、「Reserved」、または「Spot」が含まれます。残りのレコードについては、purchase_option 列に「OnDemand」と表示されます。

列名	説明
region	AWS サービスをホストする地理的エリア。このフィールドを使用して、特定のリージョンにわたって使用量を分析できます。 例: eu-west-3、us-west-1、us-east-1、ap-northeast-2、sa-east-1 サービス: • Amazon EC2 • AWS Certificate Manager • Amazon S3 • Amazon RDS • Amazon DynamoDB • 完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。
ri_sp_arn	Savings Plan またはリザーブドインスタンスの一意の ID。通常、 arn: aws: savingsplans: <region>:<account-id>:savingsplan/<savings-plan-id>または の形式に従いますarn: aws:ec2:<region>:<account>reserved-instances/<reserved-instance-id>。

列名	説明
ri_sp_trueup	これは、明細項目の種類に基づくコストの調整です。調整額は、その期間に発生した手数料の混合されていない前払い費用の合計と、償却コストを使用した場合にその期間に適用される前払い費用の小部分との差額を表します。調整は、次のように決定されます。 • 明細項目タイプが「SavingsPlanRecurringFee」の場合、調整額は、Savings Plan の請求期間における償却済み前払い契約のマイナスになります。 • 明細項目タイプが「RIFee」の場合、調整額は、予約の請求期間中の償却前払い料金のマイナスとなります。 • その他のすべての種類の明細項目では、調整額は 0 です。
ri_sp_upfront_fee	前払い料金とは、特定のタイプのリザーブドインスタンスまたは Savings Plans を選択したときに支払う最初の支払いを指します。
service	AWS サービスの名前。 例: AmazonVPC、AmazonRDS、AmazonRoute53 など。

列名	説明
tenancy	Amazon EC2 インスタンスで許可されているテナンシーのタイプ。 例: 専用、リザーブド、共有、NA、ホスト サービス: • Amazon EC2 • Amazon ECS
unblended_cost	UnblendedCost は、UsageAmount に UnblendedRate を掛けたものです。
usage_date	明細項目の開始日時は UTC で表されます。形式は `YYYY-MM-DDTHH:mm:ssZ` です。 例: 2023-10-01T00:00:00.000Z
usage_quantity	指定した期間に発生した使用量。サイズ柔軟なリザーブドインスタンスについては、代わりに reservation/TotalReservedUnits 列を使用します。 Note 特定のサブスクリプション料金では、UsageAmount が 0 になります。
usage_type	この明細項目の使用状況の詳細。例えば、USW2-BoxUsage:m2.2xlarge は米国西部 (オレゴン) リージョンの M2 ハイメモリダブルエクストララージインスタンスを表します。

炭素排出量

炭素排出量表には、推定炭素排出量が記載されています。炭素排出量データの詳細なアカウントレベルとリージョンの詳細度を提供します。Amazon S3 への自動月別配信を CSV 形式または Parquet 形式で設定できるため、既存のビジネスインテリジェンスツールやレポートシステムとの統合が簡単になります。詳細については、AWS 「 Billing User Guide」の「Viewing your carbon footprint」を参照してください。

炭素排出量の SQL テーブル名は ですCARBON_EMISSIONS。

履歴データ

設定後 24 時間以内に最大 38 か月の履歴データを受け取るため、手動によるデータ収集なしでベースライン分析と傾向レポートを実行できます。

テーブル設定

炭素排出量テーブルのテーブル設定はありません。

アクセス許可

Customer Carbon Footprint Tool または Carbon emissions テーブルのデータにアクセスするには、IAM アクセス許可 が必要ですsustainability:GetCarbonFootprintSummary。

モデルバージョン

炭素排出量を計算する方法は、使用量をよりよく反映し、炭素会計のベストプラクティスに合わせて、時間の経過とともに進化します。エクスポートは、階層順に「model_version=Y/」と「usage_period=YYYY-MM/」で分割されます。エクスポートが保存されている「model_version」パーティションは、そのエクスポートの生成に使用されるモデルバージョンに対応し、「usage_period」パーティションは炭素排出量が生成された日付に対応します。この構造により、パーティション名を表示して、古いモデルと新しいモデルでデータを区別できます。

新しいモデルバージョンをリリースした後、 は古いモデルと新しいモデルの両方のエクスポートを 3 か月間 AWS 配信するため、データを比較して方法論の変更に調整できます。

AWS Organizations のサポート

炭素排出量テーブルは、 AWS Organizations の一括請求機能で行った設定を継承します。一括決済が有効になっている場合、管理アカウントとメンバーアカウントにはさまざまな動作があります。管

理アカウントを使用している場合、炭素排出量テーブルには、管理アカウントと組織内のすべてのメンバーアカウントの推定炭素排出量データが含まれます。メンバーアカウントを使用している場合、炭素排出量テーブルには、そのメンバーアカウントの推定炭素排出量データのみが含まれます。

メンバーアカウントが新しい組織に加わるか、管理アカウントがメンバーアカウントに変換されて新しい組織に加わると、アカウントの炭素排出量データが新しい組織のエクスポートに記録されます。各管理アカウントには、その管理アカウントにリンクされた期間のメンバーアカウントのデータが含まれます。たとえば、メンバーアカウントは組織 A を離れ、その月の 15 日に組織 B に参加します。その後、メンバーアカウントはエクスポートを作成します。メンバーアカウントは組織 B への参加後にエクスポートを作成しているため、メンバーアカウントの当月の炭素排出量テーブルのエクスポートには、アカウントが組織 B のメンバーであった期間の推定炭素排出量データが含まれます。すべての炭素排出量データエクスポートと同様に、各毎月の発行には 3 か月前のデータが含まれます (たとえば、4 月の更新には 1 月のデータが含まれます) 。

メンバーアカウントが組織を離れるか、スタンドアロンアカウントに変換しても、それらのエクスポートが保存されている Amazon S3 バケットへのアクセス許可がある場合、メンバーアカウントは以前のエクスポートにアクセスできます。終了または停止されたアカウントに関連する炭素排出量は、これらのアカウントがアクティブであった期間の管理アカウントのデータエクスポートに表示されます。

詳細については、「 Billing AWS User Guide」の「Consolidated Billing for AWS Organizations」を参照してください。

炭素排出量列

列名	説明	データ型	Null 値を使用できます
last_refresh_timestamp	カーボンフットプリント値が最後に生成された時刻。UTC。形式は `YYYY-MM-DD HH:mm:ssZ` です。	timestamp	いいえ
location	region_code に対応する、使用元の人間が読める場所を記述します。例えば、米国	string	はい

列名	説明	データ型	Null 値を使用できます
	東部 (バージニア北部) などです。Amazon CloudFront などのグローバルサービスからの排出量は、 で報告されます`Global`。		
model_version	カーボンフットプリントデータが計算された方法論のバージョン。例えば、v2.0.0 などです。	string	いいえ
payer_account_id	支払いアカウントのアカウント ID。 AWS の組織の場合、これは管理アカウントのアカウント ID です。	string	いいえ
product_code	測定された製品のコード。例えば、Amazon EC2 は、Amazon Elastic Compute Cloud の製品コードです。	string	はい

列名	説明	データ型	Null 値を使用できます
region_code	リージョンは、データセンターがクラスター化されている世界中の物理的な場所です。 は、論理データセンターの各グループをアベイラビリティーゾーン (AZ) と AWS 呼びます。各 AWS リージョンは、地理的エリア内の複数の分離された物理的に分離AZs で構成されます。リージョンコード属性の名前は AWS リージョンと同じで、 AWS サービスが利用可能な場所を指定します。例えば、us-east-1 などです。	string	はい
total_mbm_emissions_unit	市場ベースの方法 (MBM) 排出量に使用される測定単位。現在サポートされている単位は、二酸化炭素換算 (MTCO2e) のメトリクストンです。	string	はい

列名	説明	データ型	Null 値を使用できます
total_mbm_emissions_value	アカウントに関連付けられた推定市場ベースメソッド (MBM) 排出量の合計。これには現在、スコープ 1 とスコープ 2 が含まれ、スコープ 3 は将来含まれます。	double	はい
usage_account_id	カーボンフットプリント値に関連付けられたアカウントのアカウント ID。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。	string	いいえ
usage_period_end	このレポートの対象となる期間の終了日を UTC で表します。形式は `YYYY-MM-DD HH:mm:ssZ` です。	timestamp	いいえ
usage_period_start	このレポートの対象となる期間の開始日を UTC で表します。形式は `YYYY-MM-DD HH:mm:ssZ` です。	timestamp	いいえ

データエクスポートの処理

次のセクションでは、データエクスポートの管理について説明します。

Amazon Athena の設定

コストと使用状況レポート (CUR) とは異なり、データエクスポートではエクスポートをクエリするように Athena を設定するための SQL ファイルは提供されていません。データエクスポートには CloudFormation テンプレートを使用するか (オプション 1 を参照)、Athena を手動で設定する必要があります (オプション 2 を参照)。

(オプション 1) CloudFormation テンプレートを使用する: CloudFormation テンプレートとデータエクスポートで Athena を設定する手順を見つけるには、「Cloud Intelligence Dashboards Framework のデータエクスポート」を参照してください。

(オプション 2) AWS Glue クローラを使用して Athena のテーブルとパーティションを構築する : Athena の CUR または炭素排出量データエクスポートを作成するときは、Apache Parquet ファイル形式を使用することをお勧めします。これにより、圧縮と列指向ストレージが改善され、Athena クエリのコストが小さくなります。上書き配信設定は、各月次パーティションに常に各ファイルのコピーが 1 つだけ含まれ、Amazon Athena でクエリを実行するときに重複する行項目が表示されないようにするために必要です。

また AWS 、 Glue クローラで Glue を使用して Athena にデータをロードすることをお勧めします。

Glue クローラを使用して Athena AWS のテーブルとパーティションを構築するには

1. 次のデータエクスポート配信オプションを使用して、CUR 2.0 または炭素排出量のエクスポートを作成します。
   - 圧縮タイプとファイル形式: Parquet - Parquet
   - ファイルバージョニング: 既存のデータエクスポートファイルを上書きする
2. Athena で、Trino SQL でノートブックエディタを使用し、作成を選択してAWS 「Glue クローラ」でテーブルを作成します。Glue クローラーワークフローを使用して、Glue クローラーを s3://<bucket-name>/<prefix>/<export-name>/data フォルダで実行するように指定し、指定されたエクスポートのすべての配信済みパーティションを Athena に自動的にロードします。
3. Glue クローラーが完了したら、Athena を使用して、Glue クローラーによって作成されたテーブルにクエリを記述できます。

Amazon Redshift の設定

Amazon Redshift は、プロビジョンドキャパシティまたはサーバーレスモデルのいずれかでアクセスできるクラウドデータウェアハウスです。Amazon Redshift は、データエクスポートからのデータを処理するための高速なクエリパフォーマンスを提供します。

現在、データエクスポートでは、コストと使用状況レポート (CUR) のようにエクスポートをクエリするように Redshift を設定するための SQL ファイルは提供されていません。ただし、エクスポートをクエリするように Athena を手動で設定することはできます。Redshift には gzip/csv 圧縮とファイル形式を使用することをお勧めします。

Redshift のセットアップの詳細については、「Amazon Redshift 入門ガイド」を参照してください。

CUR 2.0 の処理するための推奨 SQL クエリ

CUR 2.0 エクスポートデータを Amazon Athena や Amazon Redshift などのデータ分析ツールにロードした後、コストと使用状況に関するインサイトを得るためにデータを処理できます。 AWS Well-Architected Labs は、CUR の処理に使用できる CUR クエリライブラリを提供します。詳細については、「AWS CUR クエリライブラリ」を参照してください。

SQL クエリに関する次の 2 つの情報に注意してください。

- Well-Architected Labs の SQL クエリは、データエクスポートのクエリフィールドでは動作しません。データエクスポートは、集計や、これらのクエリで使用されるその他の SQL 構文の一部をサポートしていないためです。
- Well-Architected Labs の SQL クエリは、列の名前をデフォルト名から変更していない場合にのみ動作します。クエリによっては、ドット演算子を使用して製品列の一部を別の列としてクエリする必要がある場合があります。詳細については、「データクエリ – SQL クエリとテーブル設定」を参照してください。

炭素排出量データのエクスポートを処理するための推奨 SQL クエリ

payer_account_id あたりの総炭素排出量を取得するには：

```
SELECT payer_account_id, SUM(total_mbm_emissions_value) AS total_emissions
FROM "ccft-data-exports"."ccft-data-exports-data" -- change to your table name
```

```
GROUP BY payer_account_id
ORDER BY total_emissions DESC;
```

payer_account_id および product_code あたりの炭素排出量の合計を取得するには：

```
SELECT payer_account_id, product_code, SUM(total_mbm_emissions_value) AS
 total_emissions
FROM "ccft-data-exports"."ccft-data-exports-data"-- change to your table name
GROUP BY payer_account_id, product_code
ORDER BY total_emissions DESC;
```

payer_account_id および region_code あたりの総炭素排出量を取得するには：

```
SELECT payer_account_id, region_code, SUM(total_mbm_emissions_value) AS total_emissions
FROM "ccft-data-exports"."ccft-data-exports-data" -- change to your table name
GROUP BY payer_account_id, region_code
ORDER BY total_emissions DESC;
```

コストと使用状況ダッシュボードについて

コストと使用状況ダッシュボードは、Amazon QuickSight によって提供され、Cloud Intelligence Dashboards (CID) オープンソースプロジェクトから着想を得た、デプロイが容易で安全な事前構築済みのダッシュボードです。コストと使用状況ダッシュボードには、6 つの Cloud Intelligence Dashboards の 1 つである CUDOS dashboard のリソースレベルのビューを除き、概要ビジュアルのサブセットが含まれています。Cost and Usage Dashboard は、設定が簡単な AWS Billing and Cost Management コンソール機能に CUDOS ソリューションの利点をもたらし、Amazon Athena ビューや AWS Glue クローラなどの基盤となるインフラストラクチャの維持を解放します。コストと使用状況ダッシュボードは、 AWS Billing and Cost Management コンソールのデータエクスポートページから数分でデプロイできます。CID には、 AWS CloudFormation のテンプレートベースのデプロイメントが含まれます。完全な CID ソリューションの設定については、「AWS well architected labs」を参照してください。

次の表は、コストと使用状況ダッシュボードと Cloud Intelligence Dashboards (CID) の違いをまとめたものです。

機能	コストと使用状況ダッシュボード	Cloud Intelligence Dashboards
デプロイ	AWS コンソールからのシームレスなデプロイ	CloudFormation、コマンドライン、または Terraform
AWS Organizations のデプロイオプション	管理アカウントのみ	管理アカウントまたは委任されたリンクアカウント
複数の AWS Organizations 集約	いいえ	はい
コストと使用状況に関するハイレベルなインサイト	あり	あり
リソースレベルの詳細	いいえ	はい
リザーブドインスタンスと Savings Plans のインサイト	いいえ	はい

機能	コストと使用状況ダッシュボード	Cloud Intelligence Dashboards
サポートされているデータソース	コストと使用状況の概要 (ダッシュボードビュー)	コストと使用状況レポート (CUR)、Compute Optimizer、Trusted Advisor、コスト異常検知

コストと使用状況レポート (CUR) について

CUR と CUR 2.0 を使用して、 AWS コストと使用状況に関する最も詳細な情報を取得できます。Savings Plans や リザーブドインスタンス (RI) の利用状況、課金、コスト配分の分割のトラッキングなどの使用例については「Use cases」をご覧ください。

炭素排出量データのエクスポートについて

炭素排出量データのエクスポートを使用して、支払者アカウントにリンクされたすべての使用アカウントの Customer Carbon Footprint Tool から炭素排出量の見積もりを取得できます。これらの炭素排出量の計算と分類の詳細については、AWS 請求ユーザーガイドの「カーボンフットプリントの表示」を参照してください。

セキュリティおよびアクセス許可

でのクラウドセキュリティが最優先事項 AWS です。 AWS のお客様は、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 とお客様の間で共有される責任です。責任共有モデルでは、この責任が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して説明されています。

クラウドのセキュリティ： AWS は、 AWS クラウドで AWS サービスを実行するインフラストラクチャを保護する責任があります。 AWS また、 は、お客様が安全に使用できるサービスも提供します。サードパーティーの監査人は、AWS コンプライアンスプログラム の一環として、セキュリティの有効性を定期的にテストおよび検証します。 AWS コスト管理に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 「コンプライアンスプログラムによる対象範囲内のサービス」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 AWS サービスによって決まります。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、請求情報とコスト管理を使用する際に共有責任モデルを適用する方法を理解するのに役立ちます 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ために請求情報とコスト管理を設定する方法を示します。また、請求情報とコスト管理リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても説明します。

トピック

- データエクスポートの Identity and Access Management
- データエクスポートにおけるデータ保護

データエクスポートの 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するのに役立つ AWS サービスです。IAM 管理者は、誰を認証 (サインイン) し、誰に課金リソースの使用を許可 (許可を持たせる) するかを制御します。IAM は、追加料金なしで使用できる AWS サービスです。

データエクスポートを使用するには、IAM の `bcm-data-exports namespace` のアクションに対するアクセス権を IAM ユーザーに付与する必要があります。使用可能なアクションについては、次の表を参照してください。

データエクスポートアクション	説明	アクセスレベル	リソースタイプ	条件キー
CreateExport	ユーザーがエクスポートを作成できるようにし、クエリ、配信設定、スケジュール設定、コンテンツ設定を指定します。	書き込み	エクスポート テーブル	aws:RequestTag/${TagKey} aws:TagKeys
UpdateExport	ユーザーが既存のエクスポートを更新できるようにします。	書き込み	エクスポート テーブル	aws:ResourceTag/${TagKey}
DeleteExport	ユーザーが既存のエクスポートを削除できるようにします。	書き込み	エクスポート	aws:ResourceTag/${TagKey}
GetExport	ユーザーが既存のエクスポートを表示できるようにします。	読み取り	エクスポート	aws:ResourceTag/${TagKey}
ListExports	ユーザーが既存のエクスポートをすべて一覧表示できるようにします。	読み取り		

データエクスポートアクション	説明	アクセスレベル	リソースタイプ	条件キー
GetExecution	エクスポートされたデータのメタデータやスキーマなど、指定された実行の詳細をユーザーが確認できるようにします。	読み取り	エクスポート	aws:ResourceTag/${TagKey}
ListExecutions	ユーザーが、提供されたエクスポート識別子の実行をすべて一覧表示できるようにします。	読み取り	エクスポート	aws:ResourceTag/${TagKey}
GetTable	ユーザーが指定されたテーブルのスキーマを取得できるようにします。	読み取り	テーブル	
ListTables	使用可能なすべてのテーブルをユーザーが一覧表示できるようにします。	読み取り		

データエクスポートアクション	説明	アクセスレベル	リソースタイプ	条件キー
TagResource	ユーザーが既存のエクスポートにタグ付けできるようにします。	書き込み	エクスポート	aws:ResourceTag/${TagKey} aws:RequestTag/${TagKey} aws:TagKeys
UntagResource	ユーザーが既存のエクスポートのタグを解除できるようにします。	書き込み	エクスポート	aws:ResourceTag/${TagKey} aws:TagKeys
ListTagsForResource	ユーザーが既存のエクスポートに関連付けられているタグを一覧表示できるようにします。	読み取り	エクスポート	aws:ResourceTag/${TagKey}

これらのコンテキストキーの使用方法についての詳細は、「IAM ユーザーガイド」の「タグを使用した AWS リソースへのアクセスの制御」を参照してください。

次の表は、データエクスポートで利用可能なリソースタイプを示しています。

リソースタイプ	説明	ARN
エクスポート	エクスポートは CreateExport API によって作成されたリソースです。エクスポートにより、請求情報とコスト管理	arn:${Partition}:bcm-data-exports:${Region}:${Account}:export/${exportName}-{UUID}

リソースタイプ	説明	ARN
	のクエリ出力が定期的に生成されます。	
テーブル	テーブルは、ユーザーがエクスポートを使用してクエリを実行する、行と列形式のデータです。テーブルは、 によってお客様の AWS ために作成および管理されます。お客様はテーブルを削除することができません。	arn:${Partition}:bcm-data-exports:${Region}:${Account}:table/${TableName}

COST_AND_USAGE_REPORT または COST_AND_USAGE_DASHBOARD テーブルリソースのエクスポートをデータエクスポートで作成するには、IAM ユーザーが IAM 内のそれぞれの `cur` アクションに対するアクセス許可も持っている必要があります。つまり、`cur` での明示的な許可がないことや、`cur` に対する明示的な拒否を提供するサービスコントロールポリシー (SCP) がないなど、何らかの理由で IAM ユーザーが `cur` アクションの使用をブロックされた場合、その IAM ユーザーは、これら 2 つのテーブルのエクスポートの作成または更新をブロックされます。

次の表は、これら 2 つのテーブルのデータエクスポートで、どの `cur` アクションに対してどの `bcm-data-exports` アクションが必要かを示しています。

データエクスポートアクション	テーブルリソース	IAM で必要なその他のアクション
bcm-data-exports:CreateExport	COST_AND_USAGE_REPORT COST_AND_USAGE_DASHBOARD	cur:PutReportDefinition

ポリシーの例

IAM ユーザーがデータエクスポートの CUR 2.0 エクスポートにフルアクセスできるようにします。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewDataExportsTablesAndExports",
            "Effect": "Allow",
            "Action": [
                "bcm-data-exports:ListTables",
                "bcm-data-exports:ListExports",
                "bcm-data-exports:GetExport"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateCurExports",
            "Effect": "Allow",
            "Action": "bcm-data-exports:*",
            "Resource": [
                "arn:aws:bcm-data-exports:*:*:table/COST_AND_USAGE_REPORT",
                "arn:aws:bcm-data-exports:*:*:export/*"
                ]
        },
        {
            "Sid": "CurDataAccess",
            "Effect": "Allow",
            "Action":  "cur:PutReportDefinition",
            "Resource": "*"
         }
    ]
}
```

請求情報とコスト管理のデータエクスポートを使用するためのアクセスコントロールと IAM アクセス許可の詳細については、「アクセス許可の管理の概要」を参照してください。

データエクスポートにおけるデータ保護

データエクスポートでのデータ保護に 責任 AWS 共有モデルがどのように適用されるかについて説明します。

S3 セキュリティのベストプラクティス

データエクスポートは、請求情報とコスト管理データを Amazon S3 バケットに配信します。S3 バケットが安全であることを確認するために実行できる手順は多数あります。詳細については、「Amazon S3 ユーザーガイド」の「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。

S3 でのデータ暗号化

デフォルトでは、データエクスポートは、Amazon S3 マネージドキー (SSE-S3) によるサーバー側の暗号化を使用して暗号化されます。Amazon キー管理サービス (KMS) 暗号化 (SSE-KMS) を使用してエクスポートを暗号化する場合は、エクスポートの配信後に KMS による暗号化をトリガーする必要があります。詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 バケット向けのサーバー側のデフォルトの暗号化動作の設定」を参照してください。

クォータと制限

次の表は、データエクスポートにおける現在のクォータと制限について説明したものです。

クォータタイプ	クォータ値
CUR 2.0 のエクスポート数 (COST_AND_USAGE_REPORT)	5
コスト最適化の推奨事項テーブルのエクスポート数 (COST_OPTIMIZATION_RECOMMENDATIONS)	2
FOCUS 1.0 テーブルのエクスポート数 (FOCUS_1_0_AWS)	2
FOCUS 1.0 プレビューテーブルのエクスポート数 (FOCUS_1_0_AWS_PREVIEW)	2
コストと使用状況ダッシュボードテーブルのエクスポート数 (COST_AND_USAGE_DASHBOARD)	2
炭素排出量テーブルのエクスポート数 (CARBON_EMISSIONS)	2

トラブルシューティング

データエクスポートまたはコストと使用状況レポート (CUR) を使用する際、トラブルシューティングが必要な特定の問題が発生する場合があります。このトラブルシューティングセクションでは、一般的な問題を迅速かつ効率的に解決するための推奨事項を紹介します。

トピック

- 一般的なトラブルシューティング
- CUR 2.0 のトラブルシューティング
- コストと使用状況ダッシュボードのトラブルシューティング
- コストと使用状況レポートのトラブルシューティング
- 炭素排出量データエクスポートのトラブルシューティング

一般的なトラブルシューティング

トピック

- エクスポートが異常なのはなぜですか。
- SQL ステートメントがデータエクスポートで受け入れられないのはなぜですか。
- データエクスポート内で Athena を設定するための事前定義された SQL スクリプトが見つかりません。
- エクスポートパーティションの 1 つが空なのはなぜですか。
- Amazon S3 バケットにレポートファイルがないのはなぜですか？

エクスポートが異常なのはなぜですか。

「異常な」エクスポートとは、最後に Amazon S3 バケットに更新を配信しようとしたときにエラーが発生したエクスポートです。「異常」メッセージにカーソルを合わせるか、`GetExport` API を呼び出すと、以下のいずれかのエラーメッセージが表示されることがあります。

- データエクスポートに関する問題
  - 権限が不十分: これは、データエクスポートがエクスポートファイルを S3 バケットに配信できなかったことを意味します。これは、「Setting up an Amazon S3 bucket for data exports」に記載されている権限で S3 バケットポリシーを更新することで修正できます。

- 請求所有者の変更： このエラーは、 AWS アカウントが新しい組織に移動するか AWS 、Organizations で組織を離れるときに発生する可能性があります。これは、組織内にいて、 AWS Billing Conductor の請求グループに属するかどうかにかかわらず、管理アカウントが変更された場合にも発生する可能性があります。この問題を解決する最善の方法は、新しい CUR を作成し、古い CUR を削除することです。アカウントが組織や請求グループを変更すべきではないと思われる場合は、アカウント管理者に連絡してください。
- 内部障害: このエラーは、データエクスポートの内部インフラストラクチャの問題が原因です。データエクスポートに影響を与えている可能性のあるサービス全体の問題に関する更新については、 AWS サービスヘルスダッシュボードを確認するか、 AWS サポートにお問い合わせください。

- QuickSight 統合に関する問題
  - SPICE 容量が不十分: このエラーは、QuickSight にコストと使用状況データを取り込むのに十分な処理能力がプロビジョニングされていないことを意味します。SPICE 容量を増やす方法については、「SPICE メモリ容量の管理」を参照してください。
  - マニフェストファイルにアクセスするための権限が不十分: S3 バケットにアクセスするために QuickSight に割り当てたサービスロールが機能しなくなりました。サービスポリシーを見直して、コストと使用状況データを保存する S3 バケットに読み取り権限が付与されていることを確認してください。
  - マニフェストファイルにアクセスしようとするとアクセスが拒否されました: お客様の IAM ロールには、このエクスポート用の QuickSight ダッシュボードが存在するかどうかを確認するためのエクスポートファイルを保存している S3 バケットへのアクセス権がありません。ダッシュボードが動作している場合と動作していない場合があります。QuickSight ダッシュボードを確認できるようにするには、エクスポートデータを保存する S3 バケットに `s3:GetObject` アクセス許可が必要です。
  - QuickSight CreateBundle が失敗しました: このエラーは、QuickSight でダッシュボードを作成できなかったことを意味します。これは、新しいサービスロールを作成した場合や、適切な権限を持たない既存のサービスロールを選択した場合に IAM ロールの伝達が遅れたことが原因と考えられます。新しいサービスロールを作成した場合は、再試行アクションを使用してください。既存のサービスロールを選択した場合は、エクスポートを削除して、新しいサービスロールを含む新しいエクスポートを作成する必要があります。
  - ダッシュボードが存在しません: このエラーは、ダッシュボードが QuickSight で削除されたことを意味します。データエクスポートで既存のコストと使用状況ダッシュボードのエクスポートを削除し、再作成する必要があります。
  - QuickSight アカウントが存在しません: このエラーは、QuickSight アカウントが削除されたことを意味します。ダッシュボードを再度使用するには、QuickSight アカウントを再作成する必要

があります。QuickSight アカウントを再作成したら、データエクスポートで既存のコストと使用状況ダッシュボードのエクスポートを削除し、再作成する必要があります。

SQL ステートメントがデータエクスポートで受け入れられないのはなぜですか。

データエクスポートは、主に列選択と行フィルターに重点を置いた SQL 構文の一部しかサポートしていません。SQL ステートメントでは、関連するキーワードと演算子のみを使用していることを確認してください。詳細については、「Data query」を参照してください。

データエクスポート内で Athena を設定するための事前定義された SQL スクリプトが見つかりません。

コストと使用状況レポート (CUR) とは異なり、データエクスポートではエクスポートをクエリするように Athena を設定するための SQL ファイルは提供されていません。データエクスポートに CloudFormation テンプレートを使用するか、Athena を手動で設定する必要があります。詳細については、「Amazon Athena の設定」を参照してください。

エクスポートパーティションの 1 つが空なのはなぜですか。

エクスポートがほとんどのアプリケーションが処理できるサイズより大きい場合、 はレポートを複数のファイルに AWS 分割します。エクスポートの更新が以前のエクスポートよりも小さく、「上書き」モードを使用している場合、 は不要なパーティションを空のデータで AWS 上書きします。エクスポートマニフェストには、データのあるパーティションのみが一覧表示されます。レポートのマニフェストファイルを確認して、取り込む必要のない空のファイルを見つけてください。

Amazon S3 バケットにレポートファイルがないのはなぜですか？

Amazon S3 バケットポリシーが、billingreports.amazonaws.com サービスにバケットにファイルを配置するアクセス許可を付与していることを確認します。必要なバケットポリシーの詳細については、「データエクスポート用に Amazon S3 バケットを設定する」または「コストと使用状況レポート用に Amazon S3 バケットを設定する」を参照してください。

CUR 2.0 のトラブルシューティング

トピック

- CUR で使用可能な一部の列が CUR 2.0 に表示されないのはなぜですか？

- 従来のコストと使用状況レポートはどうなりますか。廃止されますか？
- CUR 2.0 のエクスポートを作成すると、レガシー CUR に影響がありますか？
- データエクスポートと CUR テーブルを使用する IAM アクセス許可があっても、CUR 2.0 のエクスポートを作成できないのはなぜですか？
- レガシー CUR 列と同じ CSV 形式でデータエクスポートを作成しようとすると、「無効な QueryStatement」エラーが発生します。このエラーを解消するにはどうすればよいですか?
- データエクスポート CUR 2.0 に移行した後、レガシー CUR エクスポートと CUR 2.0 エクスポートを同時に取得できますか?
- CUR 2.0 のエクスポートを作成しようとすると、「このアカウントはこのテーブルに対してエクスポートを作成できません」というエラーが表示されます。CUR 2.0 エクスポートを作成できないのはなぜですか。

CUR で使用可能な一部の列が CUR 2.0 に表示されないのはなぜですか？

CUR 2.0 では、4 つの列タイプが 4 つの個別の列にネストされるようになりました。ネストされた列は`product`、`discount`、`resource_tag`、`cost_category` になります。

レガシー CUR では、これらの文字列で始まる名前の列が数百ある可能性があります。バリエーションは、お客様の AWS サービスの使用や との契約によって異なります AWS。このスキーマ設計の結果、まばらに埋められることが多かった数百の列が作成されます。列の変動性により、課金スキーマが原因で SQL クエリに問題が発生する可能性もあります。

その結果、異なる列間で異なる可能性のある AWS 列が、これら 4 つの列にネストされました。一般的に使用される一部の製品列はネストされませんでした。

CUR 2.0 エクスポートの CUR のスキーマは、SQL のドット演算子を使用して再作成できます。その方法については、「Migrating from CUR to Data Exports CUR 2.0」を参照してください。

従来のコストと使用状況レポートはどうなりますか。廃止されますか？

現在、レガシー CUR を廃止する予定はありません。ただし、データエクスポートの CUR 2.0 では、整合性のあるスキーマ、ネストされたデータ、追加の列 (`bill_payer_account_name` と `line_item_usage_account_name`) といった改善点がいくつかあるため、CUR 2.0 への移行をお勧めします。

予定日は決定していませんが、最終的にコンソールの[レガシーページ]下にある[コストと使用状況レポート]を廃止する予定です。ただし、レガシー CUR を作成、更新、削除するための同じ機能はすべて、[データエクスポート]のコンソールページから利用できます。

> **Note**
>
> 別のレガシー請求機能である詳細な請求レポート (DBR) は、後日廃止される可能性があります。2019 年 7 月 8 日以後、この機能は新しいお客様にはご利用いただけません。

CUR 2.0 のエクスポートを作成すると、レガシー CUR に影響がありますか？

CUR と CUR 2.0 は 2 つの異なるレポートです。CUR 2.0 を作成する場合、既存の CUR 設定には影響しません。設定に基づいて、レガシー CUR と CUR 2.0 から選択できます。

データエクスポートと CUR テーブルを使用する IAM アクセス許可があっても、CUR 2.0 のエクスポートを作成できないのはなぜですか？

`cur:PutReportDefinition` の IAM アクセス許可も持っていることを確認してください。

レガシー CUR 列と同じ CSV 形式でデータエクスポートを作成しようとすると、「無効な QueryStatement」エラーが発生します。このエラーを解消するにはどうすればよいですか?

現在、列の名前を「/」などの特殊文字に変更して、CSV 形式のレガシー CUR 列名と一致させることはできません。サポートされている文字タイプの詳細については、「SQL クエリ」を参照してください。

データエクスポート CUR 2.0 に移行した後、レガシー CUR エクスポートと CUR 2.0 エクスポートを同時に取得できますか?

はい。最大 10 個のレガシー CUR エクスポートと 5 個の CUR 2.0 エクスポートを同時に作成できます。

CUR 2.0 のエクスポートを作成しようとすると、「このアカウントはこのテーブルに対してエクスポートを作成できません」というエラーが表示されます。CUR 2.0 エクスポートを作成できないのはなぜですか。

レガシー CUR とは異なり、CUR 2.0 では現在、見積もりの請求データを使用した CUR 2.0 のエクスポートの作成はサポートされていません。 AWS Billing Conductor の請求グループのメンバーであ

る場合は、見積り請求データのみを受け取ることができます。そのため、CUR 2.0 のエクスポートを作成しようとすると、このエラーメッセージが表示されます。レガシー CUR エクスポートを作成することもできます。

コストと使用状況ダッシュボードのトラブルシューティング

トピック

- コストと使用状況ダッシュボードを作成した直後にエクスポートに失敗したのはなぜですか。
- ダッシュボードにアクセスできないのはなぜですか。
- ダッシュボードを表示しようとすると、QuickSight アカウントの登録を解除するためにコンソールの管理ページに移動するのはなぜですか。
- 作成したばかりのコストと使用状況ダッシュボードにデータが表示されないのはなぜですか?
- コストと使用状況ダッシュボードに履歴データが表示されないのはなぜですか?
- QuickSight ダッシュボードのリンクがデータエクスポートコンソールページから消えたのはなぜですか。
- CUR 2.0 でリソースタグを視覚化するように Amazon QuickSight を設定するにはどうすればよいですか ?

コストと使用状況ダッシュボードを作成した直後にエクスポートに失敗したのはなぜですか。

IAM ロールの伝達が遅れたため、コストと使用状況ダッシュボードのエクスポートが失敗した可能性があります。このエクスポート用に新しいサービスロールを作成した場合、Amazon QuickSight には S3 バケットにアクセスしてダッシュボードを作成する権限がない可能性があります。エクスポートステータスに「マニフェストファイルにアクセスするための権限が不十分」というエラーが表示されたら、[エクスポート] を選択し、テーブルアクションメニューで [再試行] を選択します。

コストと使用状況ダッシュボードのエクスポート用に新しいサービスロールを作成しなかった場合は、QuickSight が使用するサービスロールを誤って指定している可能性があります。この場合は、エクスポートを削除して再作成し、コストと使用状況ダッシュボードコンソールのワークフローで新しいサービスロールを作成する必要があります。

ダッシュボードにアクセスできないのはなぜですか。

Amazon QuickSight のコストと使用状況ダッシュボードを表示するアクセス許可がないと、アクセスできない場合があります。トラブルシューティングを行うには、エクスポート名を選択してエクス

ポートを開きます。QuickSight の作成者フィールドをチェックして、誰がダッシュボードを作成したかを確認してください。ダッシュボードを閲覧するアクセス許可を与えるようユーザに依頼してください。

ダッシュボードを表示しようとすると、QuickSight アカウントの登録を解除するためにコンソールの管理ページに移動するのはなぜですか。

「Active directory」認証方法を使用している場合、このエラーが発生することがあります。コストと使用状況ダッシュボードのエクスポート名を選択すると、エクスポートの詳細が表示されます。[クイックサイトサインイン] を選択して QuickSight アカウントにサインインします。ダッシュボードを表示する権限を持っている場合は、ダッシュボードを表示できます。

作成したばかりのコストと使用状況ダッシュボードにデータが表示されないのはなぜですか?

すべてのデータがダッシュボードに入力されるまでに最大 24 時間かかることがあるため、コストと使用状況ダッシュボードに当月のデータを表示できない場合があります。コストと使用状況ダッシュボードのエクスポートの状態を確認してください。エクスポートステータスが「正常」の場合は、ダッシュボードが当月のデータで更新されるまで 24 時間かかります。24 時間後にダッシュボードに当月のデータが表示されない場合は、 AWS サポートにお問い合わせください。コストと使用状況ダッシュボードの作成時間は、データエクスポートコンソールページのエクスポートとダッシュボードテーブルで確認できます。

コストと使用状況ダッシュボードに履歴データが表示されないのはなぜですか?

以下のいずれかの理由により、コストと使用状況ダッシュボードに 6 か月分の履歴データが表示されない可能性があります。

- 履歴データが存在しない： 新しいアカウントであるか、 AWS Organizations のメンバーシップが最近変更されたために 6 か月の履歴支出がないアカウントがある場合、履歴データをダッシュボードに入力することはできません。
- 履歴のバックフィルはまだ進行中: データエクスポートによる履歴データのバックフィルが完了するまでに最大 24 時間かかることがあります。SDK/CLI を使用して、このエクスポートの `ListExecutions` API でバックフィル実行が失敗したかどうか、またはまだ実行中かどうかを確認できます。もう少し待つか、`ListExecutions` を使用してバックフィルが進行中でないことを確認してください。

- 履歴のバックフィルに失敗: 内部エラーにより、履歴データのバックフィルが完了しなかった可能性があります。24 時間以上経過してもバックフィルが完了していない場合は、この結論に達することができます。または、SDK/CLI の `ListExecutions` API を使用して、このエクスポートの実行が失敗していないかどうかを確認することもできます。バックフィルが失敗したと思われる場合は、コンソールのコストと使用状況ダッシュボードを再作成してみてください。2 回目に失敗する場合は、 AWS サポートに問い合わせることをお勧めします。

QuickSight ダッシュボードのリンクがデータエクスポートコンソールページから消えたのはなぜですか。

データエクスポートコンソールページは、エクスポートがリンクされている QuickSight ダッシュボードを識別するために S3 バケット内のファイルを読み取ります。このファイルが変更または削除された場合、コンソールはこのエクスポート用のダッシュボードが存在することを認識しません。ダッシュボードはまだ QuickSight に存在しますが、リンクを再表示するにはこのファイルを修復する必要があります。

CUR 2.0 でリソースタグを視覚化するように Amazon QuickSight を設定するにはどうすればよいですか？

コストと使用状況ダッシュボード機能は、リソースタグの視覚化に対応していません。ただし、CUR 2.0 エクスポートでもリソースタグデータを受け取ることができます。タグを使用してコストと使用状況を AWS 視覚化するためにサポートされている QuickSight ダッシュボードが必要な場合は、 AWS Well-Architected Labs の CUDOS ダッシュボードを参照してください。現在はレガシー CUR のデータのみを使用していますが、今後 CUR 2.0 にも対応する予定です。

コストと使用状況レポートのトラブルシューティング

トピック

- コストと使用状況レポートのデータが他の請求情報とコスト管理機能に表示されるデータと一致しないのはなぜですか？
- レポート設定を変更した後にデータをバックフィルするにはどうすればよいですか？
- Amazon S3 のレポートファイルフォルダが名前のないフォルダに保存されるのはなぜですか？
- レポートにリソース IDsを含めるオプションを選択できないのはなぜですか？
- Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で機能しないのはなぜですか? または Amazon Athena の Amazon Redshift クエリが機能しないのはなぜですか？

- レポートに含まれる列が前月から変更された理由
- レポートの列が変更された後、クエリまたはテーブルが機能しないのはなぜですか？
- レポートにクエリを実行するにはどうすればよいですか？
- Amazon EC2 Dedicated Host の請求データはどこにありますか？
- Amazon EC2 Elastic IP アドレスの請求データを解釈するにはどうすればよいですか？
- 非ブレンドレートとブレンドレートまたはコストは、一括請求でどのように異なりますか？
- レポートの一部の明細項目でブレンドレートまたはブレンドコストが 0 であるのはなぜですか？
- レポートですべての前払いリザーブドインスタンスはどのように償却されますか？

コストと使用状況レポートのデータが他の請求情報とコスト管理機能に表示されるデータと一致しないのはなぜですか？

その他の請求とコスト管理機能 (Cost Explorer、請求詳細レポート、請求とコスト管理コンソール) では、以下の理由によりコストの表示が異なる場合があります。

- 請求機能では、コストデータをさまざまな方法で四捨五入します。
- 請求機能ではデータ更新設定が異なる場合があります。たとえば、コストと使用状況レポートで、以前に決済された請求書を、請求書の確定後に適用された返金、クレジット、またはSupport 料金で自動的に更新するかどうかを選択できます。Cost Explorer には同じ項目が自動的に反映されます。このシナリオでは、コストと使用状況レポートの自動更新を有効にしないと、コストと使用状況レポートのデータは Cost Explorer のデータと一致しません。
- 請求機能では料金を異なる方法でグループ化できます。たとえば、Billing and Cost Management コンソールの請求書ページには、データ転送料金が AWS サービス料金 内の個別の データ転送 グループとして表示されます。一方、コストと使用状況レポートとCost Explorer には、各サービスの使用タイプとしてデータ転送料金が表示されます。

これらの理由を確認しても、コストと使用状況レポートとその他の請求およびコスト管理機能との間に相違があると思われる場合は、サポートケースを開いてコストデータの確認をリクエストしてください。サポートケースでは、確認するレポート名と請求期間を必ず提供してください。 ケースを開く方法については、「エクスポートとレポートに関するヘルプの取得」を参照してください。

レポート設定を変更した後にデータをバックフィルするにはどうすればよいですか？

サポートケースを開いて、コストデータのバックフィルをリクエストします。サポートケースでは、バックフィルするレポート名と請求期間を必ず指定してください。 ケースを開く方法については、「エクスポートとレポートに関するヘルプの取得」を参照してください。

以下のシナリオでは、コストデータのバックフィルを取得できないことに注意してください。

- アカウントを作成した日より前のコストデータのバックフィルは取得できません。
- を使用し AWS Organizations 、管理アカウントとして指定されているアカウントなど、組織の構造が変更された場合、以前の組織構造でデータのバックフィルを取得することはできません。
- を使用して組織 AWS Organizations を変更した場合、現在の組織に参加する前に からデータのバックフィルを取得することはできません。

Amazon S3 のレポートファイルフォルダが名前のないフォルダに保存されるのはなぜですか？

レポートのレポートパスプレフィックス に / 文字があると、Amazon S3 バケットに名前のないフォルダが生成されます。次回のレポート更新時に名前のないフォルダを削除するには、レポート設定を編集し、レポートパスのプレフィックスから / 文字を削除します。手順については、「コストと使用状況レポートの設定の編集」を参照してください。

レポートにリソース IDsを含めるオプションを選択できないのはなぜですか？

レポートを作成するときに、リソース ID を含める オプションを選択できます。[レポートバージョニング] を [既存のレポートを上書き] に設定してレポートを作成した場合、レポートを作成した後に [リソース ID を含める] の選択を変更することはできません。リソース ID を含めるには、新しいレポートを作成して [リソース ID を含める] オプションを選択する必要があります。

Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で機能しないのはなぜですか? または Amazon Athena の Amazon Redshift クエリが機能しないのはなぜですか ?

Amazon Athena データベースと Amazon Redshift データベースでは、コストと使用状況レポートの列の形式が異なります。Amazon Athena では、列名 (line_item_normalized_usage_amount) の単語の間にアンダースコアが追加されます。Amazon Redshift では、列タイプと属性 (lineitem_normalizedusageamount) の間にアンダースコアが追加されます。Amazon Athena または Amazon Redshift の列名の形式と一致するようにクエリを変更してください。

レポートに含まれる列が前月から変更された理由

がレポート AWS に含める列は、 AWS 使用状況によって異なります。すべてのレポートには、identity/、bill/、lineItem/ というプレフィックスが付いている列が含まれています。

- identity/LineItemId
- identity/TimeInterval
- bill/InvoiceId
- bill/BillingEntity
- bill/BillType
- bill/PayerAccountId
- bill/BillingPeriodStartDate
- bill/BillingPeriodEndDate
- lineItem/UsageAccountId
- lineItem/LineItemType
- lineItem/UsageStartDate
- lineItem/UsageEndDate
- lineItem/ProductCode
- lineItem/UsageType
- lineItem/Operation
- lineItem/AvailabilityZone
- lineItem/ResourceId

- lineItem/UsageAmount
- lineItem/NormalizationFactor
- lineItem/NormalizedUsageAmount
- lineItem/CurrencyCode
- lineItem/UnblendedRate
- lineItem/UnblendedCost
- lineItem/BlendedRate
- lineItem/BlendedCost
- lineItem/LineItemDescription
- lineItem/TaxType
- lineItem/LegalEntity

他のすべての列は、毎月の AWS 使用量がそれらの列に入力するデータを生成する場合にのみ含まれます。

例えば、その月に Savings Plans を使用した場合のみ、レポートに savingsPlan/ 列が含まれます。

レポートの列が変更された後、クエリまたはテーブルが機能しないのはなぜですか？

がレポート AWS に含める列は、その月の AWS 使用状況によって異なります。レポートに含まれる列は変更される可能性があるため、レポートに基づくカスタムクエリやテーブルでは、列番号の代わりに列名を参照するのがベストプラクティスです。

レポートにクエリを実行するにはどうすればよいですか？

コストと使用状況レポートのクエリの詳細については、 AWS Well-Architected Labs ウェブサイトの「CUR クエリライブラリのヘルプ」を参照してください。

Amazon EC2 Dedicated Host の請求データはどこにありますか？

ResourceID 列で、インスタンス ID ではなく専用ホスト ID を探します。Dedicated Hosts は Dedicated Host の実行時間によって計測されるため、レポートには Dedicated Host の使用状況が Dedicated Host の使用状況を、ホスト ID に関連付けられた従量制時間別に表示します。

Amazon EC2 Elastic IP アドレスの請求データを解釈するにはどうすればよいですか？

Amazon EC2 Elastic IP アドレスは集計して計測されます。つまり、レポートの各行項目が個々の Elastic IP アドレスに対応しているわけではありません。各行項目は課金対象時間の合計を表します。実行中のインスタンスに 1 つの Elastic IP アドレスを無料で割り当てることができます。インスタンスに割り当てる Elastic IP アドレスが追加されるたびに、1 時間ごとに比例配分して課金されます。さらに、 は割り当てられていない Elastic IP アドレスに対して時間単位の料金を AWS 請求します。

非ブレンドレートとブレンドレートまたはコストは、一括請求でどのように異なりますか？

の一括請求では AWS Organizations、非ブレンドレートとブレンドレートまたはコストが、組織内のスタンドアロンアカウントとリンクされたアカウントのアカウントの使用コストを理解するのに役立ちます。一部のサービスでは、使用量が増えるにつれて単価を下げることができる価格帯が提供されています。は組織内のサービスのすべての使用量を AWS 集計するため、個々のアカウントは、使用量が組織の毎月の使用量に集計されると、より早く低価格の階層にアクセスする可能性があります。

非ブレンドレートは、個々のアカウントのサービス使用量に関連する料金です。詳細項目の場合、非ブレンドコストは使用量に非ブレンドレートを掛けたものです。スタンドアロンアカウントの場合、非ブレンドコストはアカウントの使用コストになります。ブレンドレートは、組織内の総使用量に関連する料金をアカウント全体で平均したものです。詳細項目の場合、ブレンドコストは使用量にブレンドレートを掛けたものです。ブレンドコストは、組織内の連結アカウントとしてのアカウントの使用状況に起因するコストです。

非ブレンド費用とブレンド費用の計算の詳細と例については、AWS Billing ユーザーガイド の「一括請求について」を参照してください。

レポートの一部の明細項目でブレンドレートまたはブレンドコストが 0 であるのはなぜですか？

リザーブドインスタンスdiscount のある Amazon EC2 品目のブレンドレートはゼロです。これらの詳細項目の場合、LineItemType は「割引使用」です。

ブレンドコストは、使用量にブレンドレートを掛けたものです。ブレンドレートまたは使用量の値が 0 の場合、ブレンドコストも 0 になります。

レポートですべての前払いリザーブドインスタンスはどのように償却されますか？

全額前払いリザーブドインスタンスは全額前払いで支払われるため、償却された費用は、関連する期間 (1 年または 3 年) にわたって分割された前払いとしてレポートに反映されます。

reservation/AmortizedUpfrontCostForUsage および reservation/EffectiveCost は、全額前払いリザーブドインスタンスで同じレートです。これは、どちらの列も、リザーブドインスタンスの前払い金をその期間の合計時間に対して等しく割ったものだからです。

RiFee が 0.00 USD であっても、レポートには全額前払いリザーブドインスタンスの RiFee 明細項目が入力されていることが予想されます。これらの項目は、その月の定期的な時間単位のコストを表し、他の列には追加の使用量データがあります。すべてのリザーブドインスタンスは RiFee の明細項目を生成します。

炭素排出量データエクスポートのトラブルシューティング

トピック

- データエクスポートと CUR 2.0 テーブルを使用する IAM アクセス許可があっても、炭素排出量テーブルのエクスポートを作成できないのはなぜですか？
- 組織内の一部のメンバーアカウントの炭素排出量データが表示されないのはなぜですか？
- S3 バケット内のいずれかのファイルが空であるのはなぜですか？
- 使用状況データがある場合、一部のリージョンとサービスで S3 エクスポートに炭素排出量ゼロが表示されるのはなぜですか？
- 過去のデータバックフィルは、炭素排出量のデータエクスポートで利用できますか？
- S3 バケットに履歴データが表示されないのはなぜですか？
- レポートの設定を変更しました。データを新しい設定でバックフィルできますか？

データエクスポートと CUR 2.0 テーブルを使用する IAM アクセス許可があっても、炭素排出量テーブルのエクスポートを作成できないのはなぜですか？

Customer Carbon Footprint Tool または Carbon emissions テーブルのデータにアクセスするには、IAM アクセス許可 が必要です`sustainability:GetCarbonFootprintSummary`。

組織内の一部のメンバーアカウントの炭素排出量データが表示されないのはなぜですか？

管理 (支払者) アカウントを使用している場合は、管理アカウントとすべてのメンバー (使用状況) アカウントの炭素排出量データが炭素排出量テーブルに自動的に表示されます。追加の設定は必要ありません。

ただし、炭素排出量データには 3 か月のデータラグがあります。新しいメンバーアカウントの場合、メンバーアカウントが組織に加わったときを含むエクスポート期間まで、データは管理アカウントの炭素排出量データのエクスポートに表示されません。たとえば、1 月に新しいメンバーアカウントをリンクした場合、そのデータは 4 月のエクスポートに最初に表示されます。

同様に、メンバーアカウントが組織を離れると、そのデータは削除されたエクスポート期間まで引き続き表示されます。

S3 バケット内のいずれかのファイルが空であるのはなぜですか？

アカウントに特定の月の炭素排出量データがない場合、指定された炭素モデルバージョンと使用期間のファイルが S3 バケットに届きますが、ファイルは空になります。

使用状況データがある場合、一部のリージョンとサービスで S3 エクスポートに炭素排出量ゼロが表示されるのはなぜですか？

総炭素排出量がゼロと表示されている場合は、表示しきい値である 0.0000005 MTCO2e よりも低いことを意味します。

過去のデータバックフィルは、炭素排出量のデータエクスポートで利用できますか？

はい。エクスポートを作成すると、最初の配信で最大 38 か月の履歴データと、その後は毎月 1 か月のデータが届きます。アカウントが過去 38 か月以内に作成された場合は、アカウント作成日以降の炭素排出量の見積もりを受け取ります。

S3 バケットに履歴データが表示されないのはなぜですか？

次のいずれかの理由で、S3 バケットに履歴データがない可能性があります。

- 履歴データが存在しない： 新しいアカウントであるか、 AWS Organizations のメンバーシップが最近変更されたために、過去の炭素排出量の見積もりがないアカウントがある場合、S3 バケット

に履歴データを入力することはできません。アカウントが過去 38 か月以内に作成されている場合は、アカウントがアクティブになっている期間全体の炭素排出量の見積もりを受け取ります。

- 履歴のバックフィルはまだ進行中: データエクスポートによる履歴データのバックフィルが完了するまでに最大 24 時間かかることがあります。SDK/CLI を使用して、このエクスポートの `ListExecutions` API でバックフィル実行が失敗したかどうか、またはまだ実行中かどうかを確認できます。もう少し待つか、`ListExecutions` を使用してバックフィルが進行中でないことを確認してください。
- 履歴のバックフィルに失敗: 内部エラーにより、履歴データのバックフィルが完了しなかった可能性があります。24 時間以上経過してもバックフィルが完了していない場合は、この結論に達することができます。または、SDK/CLI の `ListExecutions` API を使用して、このエクスポートの実行が失敗していないかどうかを確認することもできます。バックフィルが失敗したと思われる場合は、新しいエクスポートを作成してみてください。2 回目に失敗する場合は、 AWS サポートに問い合わせることをお勧めします。

レポートの設定を変更しました。データを新しい設定でバックフィルできますか？

いいえ。データのバックフィルは現在サポートされていません。

従来のコストと使用状況レポート

このセクションでは、従来のコストと使用状況レポート機能について説明します。

データエクスポートは、詳細なコストと使用状況データを受け取るための新しく推奨される方法です AWS。詳細については、「Migrating from CUR to CUR 2.0 in Data Exports」を参照してください。

AWS コストと使用状況レポートとは

AWS コストと使用状況レポート (AWS CUR) には、利用可能な最も包括的なコストと使用状況データのセットが含まれています。コストと使用状況レポートを使用して、所有している Amazon Simple Storage Service (Amazon S3) バケットに AWS 請求レポートを発行できます。コストを時間、日、または月別、製品または製品リソース別、または自分で定義したタグ別に分類したレポートを受け取ることができます。 は、バケット内のレポートをカンマ区切り値 (CSV) 形式で 1 日 1 回 AWS 更新します。Microsoft Excel または Apache OpenOffice Calc などのスプレッドシートソフトウェアを使用してレポートを表示することも、Amazon S3 API を使用するアプリケーションからレポートにアクセスすることもできます。

AWS コストと使用状況レポートは AWS 、使用状況を追跡し、アカウントに関連する推定請求額を提供します。各レポートには、 AWS アカウントで使用する AWS 製品、使用タイプ、オペレーションの一意の組み合わせごとに明細項目が含まれます。 AWS コストと使用状況レポートをカスタマイズして、時間、日、または月別に情報を集計できます。

AWS コストと使用状況レポートは、次のことを実行できます。

- Amazon S3 バケットにレポートファイルを配信する
- レポートを 1 日に 3 回まで更新する
- CUR API AWS リファレンスを使用してレポートを作成、取得、削除する

コストと使用状況レポートの仕組み

コストと使用状況レポートを作成すると、 は指定した Amazon S3 バケットにレポート AWS を送信します。 AWS は、料金が確定されるまで、少なくとも 1 日に 1 回レポートを更新します。

レポートファイルは、1 つの .csv ファイル、または複数の .csv ファイルと 1 つのマニフェストファイルで構成されます。Amazon Athena、Amazon Redshift、または QuickSight との統合用にレポートデータを設定できます。

レポートタイムライン

レポートを作成した後、 が最初のレポートを Amazon S3 バケットに配信 AWS するまでに最大 24 時間かかることがあります。 Amazon S3

配信が開始されると、 は少なくとも 1 日に 1 回レポートファイル AWS を更新します。指定された月の各レポート更新は累積的に行われるため、レポートの各バージョンには、その月までのすべての請求データが含まれます。月間に受け取るレポート更新は推定であり、引き続き AWS のサービスを使用することで、料金は変わる場合があります。

> **Note**
>
> AWS サービスによって使用量ベースの請求情報が異なるタイミングで提供されるため、特定の時間または日の更新が異なるタイミングで着信することがあります。

AWS は、請求期間が終了するまで以前のレポートに基づいて構築されます。 は、月末に請求書を発行した後、レポートの使用料 AWS を確定します。レポートの請求期間が終了すると、 は翌月の新しいレポートを生成し、前のレポートの情報は AWS 生成されません。

レポートが確定したら、 がその月の使用量に返金、クレジット、または AWS サポート料金 AWS を適用すると、レポートを更新 AWS できます。デベロッパー、ビジネス、エンタープライズサポートは最終的な使用料に基づいて計算されるため、前月のコストと使用状況レポートの毎月 6 日または 7 日に反映されます。 は、契約または契約の条件に基づいてクレジットまたは返金 AWS を適用します AWS。

レポートファイル

レポートは Amazon S3 バケットに保存された 1 つの .csv ファイルまたは複数の .csv ファイルのコレクションです。レポートが生成するファイルの数は、レポートのバージョニングの選択とレポートのサイズによって異なります。

レポートを作成するときに、新しいレポートバージョンを作成するか、更新ごとに既存のレポートバージョンを上書きするかを選択できます。新しいレポートバージョンを作成する場合は、更新のたびにレポートにより多くのファイルが生成されます。

各レポートのサイズは 1 ギガバイトを超える場合があり、デスクトップスプレッドシートアプリケーションの容量を超えると、一部の行が表示されないことがあります。レポートがほとんどのアプ

リケーションが処理できる (約 100 万行) より大きい場合、 はレポートを Amazon S3 バケットの同じフォルダに保存されている複数のファイルに AWS 分割します。

AWS は、個別のファイルに返金も生成します。 は、毎月の請求書の解約後に返金 AWS を発行します。

レポートファイル、ファイル命名規則、バージョニングの詳細については、「レポートのバージョンを確認する」を参照してください。

レポート列

各レポートには、 AWS コストと使用状況の詳細を含む複数の列が含まれています。がレポート AWS に含める列は、その月の使用状況によって異なります。

すべてのレポートには、identity/、bill/、および lineItem/ プレフィックスを持つ列が含まれます。他のすべての列は、毎月の AWS 使用量がそれらの列に入力するデータを生成する場合にのみ含まれます。

例えば、その月に Savings Plans を使用した場合のみ、レポートに savingsPlan/ 列が含まれます。

レポートの列の詳細については、「データディクショナリ」を参照してください。

レポートの使用

Amazon S3 コンソールからレポートをダウンロードしたり、Amazon Athena を使用してレポートをクエリしたり、レポートを Amazon Redshift または QuickSight にアップロードしたりできます。

- Amazon S3 バケットを作成し、Athena を使用してデータのクエリを実行する方法の詳細については、「Amazon Athena を使用したコストと使用状況レポートのクエリ」を参照してください。
- Amazon Redshift へのアップロードの詳細については、「Amazon Redshift へのレポートデータのロード」を参照してください。
- QuickSight へのアップロードの詳細については、「」を参照してくださいAmazon QuickSight へのレポートデータのロード。

コストと使用状況レポートを作成する

> **Note**
>
> データエクスポートでは、コストと使用状況レポート (CUR) 2.0 のエクスポートを作成できます。これは、詳細なコストと使用状況データを受け取るための新しく推奨される方法です

AWS。詳細については、「Migrating from CUR to CUR 2.0 in Data Exports」を参照してください。

請求情報とコスト管理コンソールのコストと使用状況レポートページを使用して、コストと使用状況レポートを作成できます。次のセクションでは、コストと使用状況レポートの開始方法について説明します。

トピック

- コストと使用状況レポート用に Amazon S3 バケットをセットアップする
- レポートの作成

コストと使用状況レポート用に Amazon S3 バケットをセットアップする

請求レポートを受信するには、レポートを受信して保存するための Amazon S3 バケットが AWS アカウントに必要です。請求コンソールでコストと使用状況レポートを作成するときに、所有している既存の Amazon S3 バケットを選択することも、新しいバケットを作成することもできます。いずれの場合でも、以下のデフォルトバケットポリシーの適用を確認して確定するように求められます。Amazon S3 コンソールでこのポリシーを編集したり、コストと使用状況レポートを作成した後にバケット所有者を変更したりすると、 AWS はレポートを配信できなくなります。Amazon S3 バケットに保存されている請求レポートデータは、標準の Amazon S3 レートで課金されます。詳細については、「クォータと制限」を参照してください。

コストと使用状況レポートの作成時に、次のポリシーがすべてのバケットに適用されます。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "billingreports.amazonaws.com"
            },
            "Action": [
                "s3:GetBucketAcl",
                "s3:GetBucketPolicy"
            ],
            "Resource":"arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
```

```
                    "aws:SourceArn": "arn:aws:cur:us-east-1:${AccountId}:definition/*",
                    "aws:SourceAccount": "${AccountId}"
                }
            }
        },
        {
            "Sid": "Stmt1335892526596",
            "Effect": "Allow",
            "Principal": {
                "Service": "billingreports.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": "arn:aws:cur:us-east-1:${AccountId}:definition/*",
                    "aws:SourceAccount": "${AccountId}"
                }
            }
        }
    ]
}
```

このデフォルトポリシーは、バケット所有者がコストと使用状況レポートのデータを読み取ることができるようにし、コストと使用状況レポートを作成したアカウントによってバケットが所有されるようにするのに役立ちます。具体的には次のとおりです。

- コストと使用状況レポートが配信されるたびに、 AWS はまず、バケットがまだレポートを設定するアカウントによって所有されているかどうかを確認します。バケットの所有権が変更されていた場合、レポートは配信されません。これにより、アカウントの請求データの安全が確保されます。このバケットポリシーにより AWS 、 (`"Effect": "Allow"`) はバケットを所有するアカウント () を確認できます`"Action": ["s3:GetBucketAcl", "s3:GetBucketPolicy"`。
- Amazon S3 バケットにレポートを配信するには、そのバケットに対する書き込みアクセス許可 AWS が必要です。そのために、バケットポリシーは、所有するバケット (`"Effect": "Allow"`) に (`"Service": "billingreports.amazonaws.com"`) レポートを配信するアクセス許可を AWS コストと使用状況レポートサービス (`"Action": "s3:PutObject"`) に付与します ()`"Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"`。

  このバケットポリシーは、配信後のコストと使用状況レポートなど、バケット内のオブジェクトを読み取りまたは削除するための AWS アクセス許可を付与しません。

- ACL が有効になっている Amazon S3 バケットの場合、 はレポートを配信するときに `BucketOwnerFullControl` ACL をさらにレポート AWS に適用します。デフォルトでは、これらのレポートなどの Amazon S3 オブジェクトは、それらを作成したユーザーまたはサービスプリンシパルのみが読み取ることができます。ユーザーまたはバケット所有者にレポートを読み取るためのアクセス許可を付与するには、 AWS は `BucketOwnerFullControl` ACL を適用する必要があります。ACL はこれらのレポートに対する `Permission.FullControl` をバケット所有者に付与します。ただし、ACL を無効にし、Amazon S3 バケットポリシーを使用してアクセスを制御することをお勧めします。Amazon S3 はデフォルト設定を変更しており、新しく作成されたバケットでは、ACL はデフォルトで無効になっていることに注意してください。詳細については、「バケットのオブジェクト所有権のコントロールと ACL の無効化」を参照してください。

コストと使用状況レポートの請求コンソールで 無効なバケット エラーが生じた場合、レポートのセットアップ後にこのポリシーとバケット所有権が変更されていないことを確認する必要があります。

レポートの作成

> **Note**
>
> データエクスポートでは、コストと使用状況レポート (CUR) 2.0 のエクスポートを作成できます。これは、詳細なコストと使用状況データを受け取るための新しく推奨される方法です AWS。詳細については、「Migrating from CUR to CUR 2.0 in Data Exports」を参照してください。

Billing and Cost Management コンソールのコストと使用状況レポートページを使用して、コストと使用状況レポートを作成できます。個々の AWS アカウントに対して最大 10 個のレポートを作成できます。

> **Note**
>
> が Amazon S3 バケットへのレポートの配信を開始する AWS までに最大 24 時間かかる場合があります。 Amazon S3 配信が開始されると、 は AWS コストと使用状況レポートファイルを少なくとも 1 日に 1 回 AWS 更新します。

コストと使用状況レポートを作成するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. [レポートを作成] を選択します。
4. [レポート名] に、レポートの名前を入力します。
5. [追加コンテンツを報告] で、[リソース ID を含める] を選択して各リソースの ID をレポートに含めます。

> **Note**
> リソース ID を含めると、リソースごとに個別の明細項目が作成されます。これにより、 AWS の使用状況に基づいて、コストと使用状況レポートのファイルサイズが大幅に増加する可能性があります。

6. 共有リソースの詳細なコストと使用状況を含めるには、[コスト配分データの分割] を選択します (Amazon ECS と Amazon EKS)。

> **Note**
> コスト配分データの分割を含めると、リソース (つまり ECS タスクと Kubernetes ポッド) ごとに、個別の明細項目が作成されます。これにより、 AWS 使用状況に基づいて、コストと使用状況レポートファイルのサイズが大幅に増加する可能性があります。

7. データ更新設定で、 が請求書の確定後にアカウントに返金、クレジット、またはサポート料金 AWS を適用する場合に、 AWS コストと使用状況レポートを更新するかどうかを選択します。レポートが更新されると、新しいレポートが Amazon S3 にアップロードされます。
8. [次へ] を選択します。
9. [S3 バケット] で、[設定] を選択します。
10. [S3 バケットの設定] ダイアログボックスで、次のいずれかを実行します。
    - 既存バケットの選択。
    - [バケットを作成する]を選択し、バケット名を入力し、次に新しいバケットを作成するリージョンを選択します。

11. バケットポリシーを確認し、[次のデフォルトポリシーがバケットに適用されます] を選択し、[保存] を選択します。
12. [レポートパスのプレフィックス] に、レポート名に付加するレポートパスのプレフィックスを入力します。
13. [時間粒度] で、次のいずれかを選択します。
    - 時間単位: レポートの明細項目を 1 時間ごとに集計する場合に選択します。
    - 日単位: レポートの明細項目を 1 日ごとに集計する場合に選択します。
    - 月単位 レポートの明細項目を 1 か月ごとに集計する場合に選択します。
14. [レポートバージョニング] で、レポートの各バージョンでレポートの以前のバージョンを上書きするのか、以前のバージョンに加えて配信するのかを選択します。

    レポートを上書きすると、Amazon S3 ストレージコストを節約できます。新しいレポートバージョンを配信することで、時間の経過とともに請求データの監査性を向上させることができます。
15. レポートデータ統合では、コストと使用状況レポートが Amazon Athena、Amazon Redshift、または QuickSight と統合できるようにするかどうかを選択します。レポートは、以下の形式で圧縮されています。
    - Athena: parquet 形式
    - Amazon Redshift または QuickSight: .gz 圧縮
16. [次へ] を選択します。
17. レポートの設定を確認したら、[確認して完了] を選択します。

請求情報とコスト管理コンソールのコストと使用状況レポートページにいつでも戻り、レポートが最後に更新された日時を確認できます。

レポートの表示と管理

コストと使用状況レポートを表示するには、請求とコストマネジメントコンソールを使用します。Amazon S3 コンソールを使用して、レポートのファイルを表示できます。

次の手順に従って、レポートとレポートファイルを検索します。

レポートの詳細とファイルを表示するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. レポートのリストから、表示するレポートの名前を選択します。
4. レポートの詳細 ページでは、レポートの設定を表示できます。
5. レポートのファイルを表示する場合、レポートの詳細 ページのレポートパスのプレフィックスに注意してください。
6. Amazon S3 バケットの下に表示されているバケット名を選択します。このリンクにより、このバケットが Amazon S3 コンソールで開きます。
7. バケット内のオブジェクトのリストから、ステップ 5 で注目した レポートパスのプレフィックス の最初の部分を名前に持つフォルダを選択します。たとえば、レポートパスのプレフィックスが **`example-report-prefix/example-report-name`** の場合、example-report-prefix という名前のフォルダーを選択します。
8. フォルダー内のオブジェクトのリストから、ステップ 5 で注目したレポートパスのプレフィックス の 2 番目の部分を名前に持つフォルダーを選択します。たとえば、レポートパスのプレフィックス が **`example-report-prefix/example-report-name`** の場合、example-report-name という名前のフォルダーを選択します。このフォルダーにはレポートファイルが含まれます。

最新のレポートバージョンを表示する

AWS は、料金が確定されるまで、少なくとも 1 日に 1 回コストと使用状況レポートを更新します。レポートを作成するときに、新しいレポートバージョンを作成するか、更新ごとに既存のレポートバージョンを上書きするかを選択できます。

更新のたびに新しいレポートバージョンを作成するようにレポートを設定した場合は、マニフェストファイルの assemblyId を使用して最新のレポートファイルを検索してください。

複数のレポートバージョンがある場合に最新のレポートファイルを表示するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. レポートのリストから、表示するレポートの名前を選択します。

4. レポートの詳細 ページで、レポートパスのプレフィックス に注意してください。
5. Amazon S3 バケットの下に表示されているバケット名を選択します。このリンクにより、このバケットが Amazon S3 コンソールで開きます。
6. バケット内のオブジェクトのリストから、ステップ 4 で注目した レポートパスのプレフィックス の最初の部分を名前に持つフォルダを選択します。たとえば、レポートパスのプレフィックス が **`example-report-prefix/example-report-name`** の場合、example-report-prefix という名前のフォルダーを選択します。
7. フォルダー内のオブジェクトのリストから、ステップ 4 で注目した レポートパスのプレフィックスの 2 番目の部分を名前に持つフォルダーを選択します。たとえば、レポートパスのプレフィックスが **`example-report-prefix/example-report-name`** の場合、example-report-name という名前のフォルダーを選択します。
8. 最新の請求期間を含む名前のフォルダーを (YYYYMMDD-YYYYMMDD 形式で) 開きます。
9. **`example-report-name`**-Manifest.json ファイルを開きます。
10. マニフェストファイルの上部の assemblyId に注目してください。assemblyId の値は、最新のレポートファイルが含まれるフォルダーの名前に対応しています。
11. Amazon S3 コンソールページに戻ります。このページには、最新の請求期間の名前のフォルダが表示されています。
12. ステップ 10 で注目した assemblyId という値を名前に持つフォルダを開きます。たとえば、assemblyId の値が**`20210129T123456Z`**の場合、20210129T123456Z/ という名前のフォルダーを開きます。このフォルダーには、最新のレポートファイルが含まれています。

確定レポートの表示

月末に請求書を発行すると、 はレポートの使用料 AWS を確定します。レポートの明細項目が確定しているかどうかを判断するには、Bill/InvoiceId 列を確認してください。明細項目が最終の場合、 bill/InvoiceId 列に AWS 請求書 ID が入力されます。明細項目がまだ確定していない場合、bill/InvoiceId 列は空白になっています。

レポート全体が確定しているかどうかを確認するには、bill/InvoiceId 列を確認してください。レポートが確定していれば、bill/InvoiceId 列に請求書 ID の値が入力されています。レポートがまだ確定していなければ、bill/InvoiceId 列は空白になっています。

> **Note**
>
> レポートが確定したら、 がその月の使用量に返金、クレジット、またはサポート料金 AWS を適用すると、レポートを更新 AWS できます。デベロッパー、ビジネス、エンタープライ

ズサポートは最終的な使用料に基づいて計算されるため、前月のレポートの毎月 6 日または 7 日に反映されます。 は、契約または契約の条件に基づいてクレジットまたは返金 AWS を適用します AWS。

レポートのバージョンを確認する

AWS は、料金が確定されるまで、少なくとも 1 日に 1 回コストと使用状況レポートを更新します。レポートを作成するときに、新しいレポートバージョンを作成するか、更新ごとに既存のレポートバージョンを上書きするかを選択できます。

レポートファイルには、1個または複数のcsv ファイルと1個のマニフェストファイルが含まれます。レポートには、Amazon Athena、Amazon Redshift、または QuickSight とのデータの統合をサポートする追加のファイルを含めることもできます。

以下のセクションでは、選択したレポートのバージョン管理に基づくファイル構成と命名規則について説明します。

コストと使用状況レポートの配信タイムライン

レポート期間中、 は請求期間が終了するまで、以前のレポートで report. AWS builds AWS を更新するたびに、新しいレポートと新しいマニフェストファイルを AWS 配信します。レポートの請求期間が終了すると、 は前のレポートの情報を含まない新しいレポート AWS を生成します。

コストと使用状況レポートの新しいバージョンの作成

以前のコストと使用状況レポートを保持する場合、CUR AWS は次の Amazon S3 の組織および命名規則を使用します。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<assemblyId>/<example-report-name>-<file-number>.csv.<zip|gz>
```

- `report-prefix` = レポートに割り当てるプレフィックス。
- `report-name` = レポートに割り当てる名前。
- `yyyymmdd-yyyymmdd` = レポートの対象となる日付の範囲。レポートは、日付範囲の終了時に確定されます。
- `assemblyId` = レポートが更新されるたびに が AWS 作成する ID。
- `file-number` = 更新に大きなファイルが含まれている場合、複数のファイルに分割 AWS される可能性があります。`file-number` で単一更新内の複数のファイルが追跡されます。

- csv = レポートファイルの形式。
- zip または gz = レポートファイルに適用される圧縮タイプ。

たとえば、レポートは以下のファイルの集まりとして配信される場合があります。

```
<example-report-prefix>/<example-report-name>/20160101-20160131/<123456789>/<example-report-name>-<1>.csv.<zip>
<example-report-prefix>/<example-report-name>/20160101-20160131/<123456789>/<example-report-name>-<2>.csv.<zip>
<example-report-prefix>/<example-report-name>/20160101-20160131/<123456789>/<example-report-name>-<3>.csv.<zip>
<example-report-prefix>/<example-report-name>/20160101-20160131/<123456789>/<example-report-name>-Manifest.json
<example-report-prefix>/<example-report-name>/20160101-20160131/<example-report-name>-Manifest.json
```

AWS は、レポート日付範囲内のすべてのレポートを同じreport-prefix/report-name/yyyymmdd-yyyymmddフォルダに配信します。 は、各レポートに一意の ID AWS を付与し、日付範囲フォルダのassemblyIdサブフォルダに配信します。レポートのサイズが大きすぎて 1 つのファイルに収まらない場合は、複数のファイルに分割されて同じ assemblyId フォルダに配信されます。

以前のレポートを保持するときのマニフェストファイルの詳細については、「コストと使用状況レポートのマニフェストファイル」を参照してください。

以前のコストと使用状況レポートの上書き

以前のコストと使用状況レポートを上書きすることを選択すると、CUR AWS は次の Amazon S3 の組織および命名規則を使用します。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-<file-number>.csv.<zip|gz>
```

- report-prefix = レポートに割り当てるプレフィックス。
- report-name = レポートに割り当てる名前。
- yyyymmdd-yyyymmdd = レポートの対象となる日付の範囲。 AWS は日付範囲の終了時にレポートを確定します。
-

- file-number = 更新に大きなファイルが含まれている場合、複数のファイルに分割 AWS される可能性があります。file-number で単一更新内の複数のファイルが追跡されます。
- csv = レポートファイルの形式。
- zip または gz = レポートファイルに適用される圧縮タイプ。

たとえば、レポートは以下のファイルの集まりとして配信される場合があります。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
<1>.csv.<zip>
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
<2>.csv.<zip><example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-
report-name>-<3>.csv.<zip>
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
Manifest.json
```

Athena の仕様

AWS CUR の作成時に Athena サポートを選択した場合、ファイルの命名規則は、形式と圧縮を除いて CUR AWS を上書きすることを選択した場合と同じです。Athena CUR ファイルは .parquet代わりに AWS を使用します。たとえば、レポートは以下のファイルの集まりとして配信される場合があります。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-
name>.parquet
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/
<cost_and_usage_data_status>
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
Manifest.json
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
create-table.sql
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/crawler-cfn.yml
```

AWS CloudFormation 仕様

は、 AWS CUR ファイルに加えて、Athena を使用して Amazon S3 データをクエリできる AWS CloudFormation スタックをセットアップするために使用できる AWS CloudFormation テンプレート AWS も提供します。 AWS CloudFormation テンプレートを使用しない場合は、提供された SQL を使用して独自の Athena テーブルを作成できます。詳細については、「Amazon Athena を使用したコストと使用状況レポートのクエリ」を参照してください。

コストと使用状況レポートのマニフェストファイル

が AWS CUR AWS を更新すると、 は Amazon Athena、Amazon Redshift、または QuickSight に使用できるマニフェストファイル AWS も作成して配信します。

マニフェストファイルは、命名規則を使用し、以下の項目を一覧表示します。

- これまでにレポートに含まれたすべての詳細列
- レポートが複数のファイルに分割された場合のレポートファイルのリスト
- レポートの対象期間およびその他の情報。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-
Manifest.json
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<assemblyId>/<example-
report-name>-Manifest.json
<example-report-prefix>/<example-report-name>/<example-report-name>/year=2018/month=12/
<example-report-name>-Manifest.json
```

コストと使用状況レポートの新しいバージョンの作成

以前のコストと使用状況レポートを保持する場合、マニフェストファイルは日付範囲フォルダと `assemblyId` フォルダの両方に配信されます。が日付範囲の新しい AWS CUR AWS を作成するたびに、日付範囲フォルダに保存されているマニフェストファイルが更新されたマニフェストファイルで上書きされます。 は、同じ更新されたマニフェストファイルを、その更新のファイルとともに`assemblyId`フォルダに AWS 配信します。`assemblyId` フォルダのマニフェストファイルは上書きされません。

以前のコストと使用状況レポートの上書き

以前の CUR AWS を上書きすると、マニフェストファイルが `month=mm`フォルダに配信されます。マニフェストファイルは、レポートファイルとともに上書きされます。

Amazon Redshift の仕様

AWS CUR で Amazon Redshift サポートのオプションを選択した場合、 はレポートを Amazon Redshift にアップロードするために必要な SQL コマンドを含むファイル AWS も作成して配信します。SQL ファイルは通常のテキストエディタで開くことができます。この SQL ファイルには次の命名規則が使用されます。

```
<example-report-prefix>/<example-report-name>/yyyymmdd-yyyymmdd/<assemblyId>/<example-report-name>-RedshiftCommands.sql
```

RedshiftCommands ファイルでコマンドを使用する場合、RedshiftManifest ファイルを開く必要はありません。

> **Important**
>
> manifest ファイルは、copy ファイルの RedshiftCommands コマンドを使用してアップロードするファイルを決定します。manifest ファイルを削除すると、RedshiftCommands ファイルのコピーコマンドで問題が生じます。

Amazon Athena の仕様

AWS CUR で Amazon Athena サポートのオプションを選択した場合、 は必要なすべてのリソースのセットアップに役立つ複数のファイル AWS も作成して配信します。 は AWS CloudFormation 、テンプレート、Athena テーブルを手動で作成するための SQL を含む SQL ファイル、および CUR 更新ステータスをチェックするための SQL AWS を含むファイルを AWS 配信します。これらのファイルでは次の命名規則が使用されます。

```
<example-report-prefix>/<example-report-name>/<example-report-name>/yyyymmdd-yyyymmdd/crawler-cfn.yml
<example-report-prefix>/<example-report-name>/<example-report-name>/yyyymmdd-yyyymmdd/<example-report-name>-create-table.sql
<example-report-prefix>/<example-report-name>/<example-report-name>/yyyymmdd-yyyymmdd/<cost_and_usage_data_status>
```

コストと使用状況レポートの設定の編集

請求情報とコスト管理コンソールのコストと使用状況レポートページを使用して、コストと使用状況レポートを編集できます。

> **Note**
>
> レポート名は編集できません。[レポートバージョニング] で [上書き] を選択すると、レポートにリソース ID、時間詳細度、またはレポートバージョニングが含まれているかどうかにかかわらず、レポート名を編集することはできません。[上書き] に設定されたレポートを削除

して、同じ名前、Amazon S3 バケット、およびパスプレフィックスで新しいレポートを作成すると、データが破損して不正確になる場合があります。

コストと使用状況レポートを編集するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. 編集するレポートを選択し、[レポートの編集] を選択します。
4. (バージョン管理されたレポートのみ) [追加コンテンツをレポート] で、[リソース ID を含める] を選択して、レポートに個々のリソースの ID を含めます。

   > **Note**
   > リソース ID を含めると、リソースごとに個別の明細項目が作成されます。これにより、 AWS の使用状況に基づいて、コストと使用状況レポートのファイルサイズが大幅に増加する可能性があります。

5. 共有リソースの詳細なコストと使用状況を含めるには、[コスト配分データの分割] を選択します (Amazon ECS と Amazon EKS)。

   > **Note**
   > コスト配分データの分割を含めると、リソース (つまり ECS タスクと Kubernetes ポッド) ごとに、個別の明細項目が作成されます。これにより、 AWS 使用状況に基づいて、コストと使用状況レポートファイルのサイズが大幅に増加する可能性があります。

6. データ更新設定で、 が請求書の確定後にアカウントに返金、クレジット、またはサポート料金 AWS を適用する場合に、 AWS コストと使用状況レポートを更新するかどうかを選択します。レポートが更新されると、新しいレポートが Amazon S3 にアップロードされます。
7. [次へ] を選択します。
8. [S3 バケット] に、レポートの配信先となる Amazon S3 バケットの名前を入力します。
9. 確認を選択します。

> **Note**
>
> バケットでは、適切なアクセス権限が有効になっている必要があります。バケットにアクセス許可を追加する方法の詳細については、Amazon Simple Storage Service ユーザーガイドのバケットとオブジェクトのアクセス許可の設定を参照してください。

10. [レポートパスのプレフィックス] に、レポート名に付加するレポートパスのプレフィックスを入力します。
11. (バージョン管理されたレポートのみ) [時間粒度] で、次のいずれかを選択します。
    - 時間単位: レポートの明細項目を 1 時間ごとに集計する場合に選択します。
    - 日単位: レポートの明細項目を 1 日ごとに集計する場合に選択します。
    - 月単位 レポートの明細項目を 1 か月ごとに集計する場合に選択します。
12. (バージョン管理されたレポートのみ) [レポートバージョニング] で、レポートの各バージョンでレポートの以前のバージョンを上書きするか、以前のバージョンに加えて配信するかを選択します。
13. レポートデータ統合では、CUR AWS が Amazon Athena、Amazon Redshift、または QuickSight と統合できるようにするかどうかを選択します。レポートは、以下の形式で圧縮されています。
    - Athena: Parquet 形式
    - Amazon Redshift または QuickSight: .gz 圧縮
14. [保存] を選択します。

のコストと使用状況レポートの使用 AWS Organizations

では AWS Organizations、管理アカウントとメンバーアカウントの両方がコストと使用状況レポートを作成できます。レポートの作成を許可または制限する IAM ポリシーは、どちらのタイプのアカウントでも同じです。

> **Note**
>
> コストと使用状況レポートを作成するアカウントは、 AWS レポートの送信先の Amazon S3 バケットも所有している必要があります。別のアカウントが所有する Amazon S3 バケットに配信するようにコストと使用状況レポートを設定することはできません。Amazon S3 バ

ケットのセットアップ要件の詳細については、「」を参照してください[コストと使用状況レポート用に Amazon S3 バケットをセットアップする](#)。

メンバーアカウントを使用したコストと使用状況レポートの管理

組織内のメンバーアカウントのコストと使用状況レポートを作成する権限を持っている場合は、メンバーアカウントのコストと使用状況のデータのみを対象とするレポートを作成できます。メンバーアカウントは、そのアカウントが現在の組織のメンバーであった期間中に、コストと使用状況に関するレポートを受け取ります。

たとえば、あるメンバーアカウントが組織 A を離れ、同月 15 日に組織 B に加入したとします。その後、メンバーアカウントはレポートを作成します。メンバーアカウントは組織 B に加入した後にレポートを作成したため、その月のメンバーアカウントのレポートには、そのアカウントが組織 B のメンバーであった期間の請求データのみが含まれます。

メンバーアカウントが新しい組織に加入すると、そのメンバーアカウントの費用と使用量が新しい組織のレポートに記録されます。これは、管理アカウントをメンバーアカウントに変換して新しい組織に加入した場合と同じ結果です。

メンバーアカウントが組織を離れたり、スタンドアロンアカウントに変換された場合でも、メンバーアカウントは、以前のレポートが保存されている Amazon S3 バケットへのアクセス権限を持っている限り、以前のレポートにアクセスできます。

コストと使用状況レポートを管理アカウントとして管理

AWS Organizations 管理アカウントの管理者で、メンバーアカウントにレポートを作成させたくない場合は、メンバーアカウントがレポートを作成できないようにするサービスコントロールポリシー (SCP) を適用できます。SCP では、メンバーアカウントによる新しいレポートの作成を禁止できますが、以前に作成されたレポートは削除されません。

> **Note**
>
> SCP はメンバーアカウントにのみ適用されます。管理アカウントがレポートを作成できないようにするには、管理アカウントのユーザーロールに添付されている IAM ポリシーを変更します。

一括請求の詳細については、AWS Billing ユーザーガイド の [AWS Organizationsの一括請求](#) を参照してください。

Amazon Athena を使用したコストと使用状況レポートのクエリ

Amazon Athena は、標準 SQL を使用して Amazon Simple Storage Service (Amazon S3) の AWS コストと使用状況レポート (AWS CUR) のデータを分析するために使用できるサーバーレスクエリサービスです。Amazon S3 これにより、 AWS CUR データをクエリするために独自のデータウェアハウスソリューションを作成する必要がなくなります。

Athena で使用する新しい Amazon S3 バケットと新しい AWS CUR レポートの両方を作成することを強くお勧めします。 AWS CUR は Athena の Apache Parquet 圧縮形式のみをサポートし、S3 バケットに保存されている以前のレポートを自動的に上書きします。

このセクションでは、コストと使用状況レポートで Athena を使用する方法について説明します。Athena サービスの詳細については、Amazon Athena ユーザーガイド を参照してください。

トピック

- AWS CloudFormation テンプレートを使用した Athena のセットアップ
- Athena の手動セットアップ
- Amazon Athena クエリの実行
- 「他のリソースへのレポートデータのロード」

Athena を使用したレポートのクエリのデモンストレーションについては、次の動画を参照してください。

Amazon Athena を使用したコストと使用状況レポートの分析

AWS CloudFormation テンプレートを使用した Athena のセットアップ

> **Important**
>
> AWS CloudFormation はクロスリージョンリソースをサポートしていません。 AWS CloudFormation テンプレートを使用する場合は、同じ AWS リージョンにすべてのリソースを作成する必要があります。リージョンでは、以下のサービスをサポートする必要があります。
>
> - AWS Lambda
> - Amazon Simple Storage Service (Amazon S3)
> - AWS Glue

- Amazon Athena

コストと使用状況レポートと Athena の統合を合理化および自動化するために、 は、Athena 統合用に設定したレポートとともに、いくつかの主要なリソースを含む AWS CloudFormation テンプレート AWS を提供します。 AWS CloudFormation テンプレートには AWS Glue 、クローラ、 AWS Glue データベース、および AWS Lambda イベントが含まれます。

を使用した Athena 統合セットアッププロセスでは、バケットに既に存在する可能性のある Amazon S3 イベントがすべて AWS CloudFormation 削除されます。これは、既存の CUR AWS レポートに対する既存のイベントベースのプロセスに悪影響を及ぼす可能性があります。Athena で使用する新しい Amazon S3 バケットと新しい AWS CUR レポートの両方を作成することを強くお勧めします。

CloudFormation テンプレートを使用して Athena 統合を自動化するには、事前に以下を実行してください。

- レポートの新しい Amazon S3 バケットを作成します。詳細については、Amazon S3 ユーザーガイドのバケットの作成を参照してください。
- Athena で使用するための新しいレポートを作成します。セットアッププロセス中に、レポートデータ統合で Athena を選択します。
- Amazon S3 バケットに最初のレポートが配信されるのを待ちます。が最初のレポートを配信 AWS するまでに最大 24 時間かかる場合があります。

Athena AWS CloudFormation テンプレートを使用するには

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. バケットのリストから、CUR AWS レポートの受信を選択したバケットを選択します。
3. レポートパスのプレフィックス (*your-report-path-prefix/*) を選択します。次に、レポート名 (*your-report-name/*) を選択します。
4. `.yml` テンプレートファイルを選択します。
5. [オブジェクトアクション]を選択し、[名前を付けてダウンロード]を選択します。
6. https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。
7. AWS CloudFormation を使用したことがない場合は、新しいスタックの作成を選択します。それ以外の場合は、[スタックを作成] を選択します。
8. [テンプレートを準備] の [テンプレートの準備ができています] を選択します。

9. [テンプレートのソース] で、[テンプレートファイルをアップロード] を選択します。
10. [ファイルを選択]を選択します。
11. ダウンロードした `.yml` テンプレートを選択し、[開く] を選択します。
12. [次へ]を選択します。
13. [スタック名] としてテンプレートの名前を入力し、[次へ] を選択します。
14. [次へ] を選択します。
15. ページの下部で、IAM リソースを作成する AWS CloudFormation 可能性があることを確認します。

    このテンプレートによって以下のリソースが作成されます。

    - 3 つのIAM ロール
    - AWS Glue データベース
    - AWS Glue クローラ
    - 2 つの Lambda 関数
    - Amazon S3 の通知
16. [スタックを作成] を選択します。

既存の Athena AWS CloudFormation テンプレートを更新するには

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. バケットのリストから、CUR AWS レポートの受信を選択したバケットを選択します。
3. レポートパスのプレフィックス (*your-report-path-prefix/*) を選択します。次に、レポート名 (*your-report-name/*) を選択します。
4. `.yml` テンプレートファイルを選択します。
5. [オブジェクトアクション]を選択し、[名前を付けてダウンロード]を選択します。
6. https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。
7. 以前に作成したスタックを選択し、[更新] を選択します。
8. [テンプレートを準備] で、[現在のテンプレートを置換] を選択します。
9. [テンプレートのソース] で、[テンプレートファイルをアップロード] を選択します。
10. [ファイルを選択]を選択します。
11. ダウンロードした `.yml` テンプレートを選択し、[開く] を選択します。

12. [次へ]を選択します。
13. [スタック詳細を指定] ページで、任意の詳細を変更し、[次へ] を選択します。
14. [次へ] を選択します。
15. ページの下部で、 が IAM リソースを作成する AWS CloudFormation 可能性があることを承認します。
16. [スタックを更新]を選択します。

Athena の手動セットアップ

AWS CloudFormation テンプレートを使用して、自分で作成するのではなく、テーブルを作成することを強くお勧めします。提供された SQL クエリは、1 か月のデータのみを対象とするテーブルを作成しますが、 AWS CloudFormation テンプレートは複数の月を含めることができ、自動的に更新されるテーブルを作成します。 AWS CloudFormation テンプレートの設定方法の詳細については、「」を参照してくださいthe section called “CloudFormation を使用した Athena のセットアップ”。

AWS CloudFormation テンプレートを使用して Athena テーブルをセットアップしない場合は、以下の手順を手動で実行します。CUR データに対して SQL AWS クエリを実行する前に、テーブルを作成する必要があります。このステップは少なくとも 1 か月に 1 回実行する必要があり、テーブルには現在の CUR AWS のデータのみが含まれます。

テーブル作成プロセスの一環として、 は CUR AWS 列名を AWS 変換します。変換プロセスの詳細については、「the section called “列名”」を参照してください。

- Athena テーブルの作成
- コストと使用状況レポートのステータステーブルの作成
- レポートパーティションのアップロード

Athena テーブルの作成

AWS には、CUR バケットにこのテーブルを作成するために実行する必要がある SQL AWS が含まれています。

Athena テーブルを作成するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/s3/ で Amazon S3 コンソールを開きます。

2. バケットのリストから、コストと使用状況レポートの送信先として指定したバケットを選択します。
3. パス *your-report-prefix*-*your-report-name*-*path-to-report* に移動します。

   正確なパスは、CUR AWS が以前のバージョンを上書きするように設定されているかどうかによって異なります。詳細については、「コストと使用状況レポートの配信タイムライン」を参照してください。
4. *my-report-name*-create-table.sql ファイルを開きます。
5. ファイルから、SQL をコピーします (CREATEから LOCATION 's3:*//your-report-prefix/your-report-name/the-rest-of-the=path*' までの部分)。Athena データベースを作成するにはデータベース名とテーブルが必要であるため、最初の行を書き留めておきます。
6. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
7. 次の SQL を [新規クエリ 1] クエリペインに貼り付けます。*<database name>.<table name>* には、コピーした SQL の最初の行にあるデータベースとテーブルの名前を使用します。

```
CREATE DATABASE <database name>
```

8. [クエリを実行]を選択します。
9. ドロップダウンメニューで、作成したデータベースを選択します。
10. [新規クエリ 1] クエリペインで、SQL ファイルから残りの SQL を貼り付けます。
11. [クエリを実行] を選択します。

テーブルを作成したら、クエリを実行する前にパーティションにロードする必要があります。詳細については、「レポートパーティションのアップロード」を参照してください。

コストと使用状況レポートのステータステーブルの作成

AWS は CUR AWS を 1 日に複数回更新します。Athena がレポートの更新中 AWS であるタイミングを に伝える方法はありません。これにより、古いデータと新しいデータを組み合わせたクエリ結果が発生する可能性があります。これを軽減するには、 がコストと使用状況レポートを更新するかどうか AWS を追跡するテーブルを作成し、そのテーブルをクエリして AWS がデータを更新するかどうかを確認します。このテーブルは 1 回のみ作成する必要があります。その後、 はテーブルを最新の状態に AWS 保ちます。

更新テーブルを作成するには

1. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
2. 次の SQL を [新規クエリ 1] クエリペインに貼り付けます。

```
CREATE EXTERNAL TABLE IF NOT EXISTS cost_and_usage_data_status(
  status STRING)
ROW FORMAT SERDE
  'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
WITH SERDEPROPERTIES (
 'serialization.format' = '1'
)
LOCATION 's3://{S3_Bucket_Name}/{Report_Key}/cost_and_usage_data_status/'
```

3. [クエリを実行] を選択します。

AWS がデータを更新しているかどうかを確認するには、Athena コンソールを使用して次の SQL クエリを実行します。

```
select status from cost_and_usage_data_status
```

レポートパーティションのアップロード

コストと使用状況レポートのデータにクエリを実行するには、Athena テーブルにデータをアップロードする必要があります。これは、 が AWS 配信する新しい CUR AWS レポートごとに実行する必要があります。

最新のレポートパーティションをアップロードするには

1. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
2. テーブル名の横にある縦の 3 つのドットを選択します。
3. パーティションのロードを選択します。

パーティションをアップロードしなければ、Athena から結果が返されないか、データが欠落していることを示すエラーメッセージが返されます。

Amazon Athena クエリの実行

データに対して Athena クエリを実行するには、まず Athena コンソールを使用して AWS がデータを更新しているかどうかを確認し、次に Athena コンソールでクエリを実行します。SQL を実行するときは、正しいデータベースがドロップダウンリストから選択されていることを確認してください。次の SQL を使用してステータスを確認できます。

```
select status from cost_and_usage_data_status
```

得られる結果は READY と UPDATING の 2 つです。ステータスが READY の場合、Athena データベースにクエリを実行できます。ステータスが UPDATING の場合、Athena は不完全な結果を返すことがあります。

AWS がデータを更新していることを確認したら、独自のクエリを実行できます。たとえば、次のクエリでは、mycostandusage_parquet というサンプルデータベースで、サービスによる毎月の当会計年度の初めから今日までのコストを示します。次のクエリは、2018 年初来のコストを示しています。年を更新して、現在の年初来の費用を確認してください。

```
SELECT line_item_product_code,
sum(line_item_blended_cost) AS cost, month
FROM mycostandusage_parquet
WHERE year='2018'
GROUP BY  line_item_product_code, month
HAVING sum(line_item_blended_cost) > 0
ORDER BY  line_item_product_code;
```

列名

Athena の列名の制限は、コストと使用状況レポート列名の制限とは異なります。つまり、CUR AWS データが Athena テーブルにアップロードされると、列名は次の変更 AWS を行います。

- 大文字の前に下線を追加
- 大文字を小文字で置換
- アルファベット以外の文字をアンダースコアで置換
- 重複する下線を削除
- 先頭および末尾のアンダースコアをすべて削除
- 列名が列名で許容される長さを超える場合、左から右の順で下線を削除

> **Note**
> これらのルールを適用すると、同じ名前の列が複数ある場合、一部のリソースタグ列には重複した names. AWS merges 列が含まれます。

例えば、列名 `ExampleColumnName` は `example_column_name` になり、列名 `Example Column Name` は `example_column_name` になります。

「他のリソースへのレポートデータのロード」

コストと使用状況レポートを Amazon Redshift と Amazon QuickSight にアップロードして、 AWS コストと使用状況を分析できます。

トピック

- Amazon QuickSight へのレポートデータのロード
- Amazon Redshift へのレポートデータのロード

Amazon QuickSight へのレポートデータのロード

コストと使用状況レポートを Amazon QuickSight にアップロードできます。

QuickSight へのアップロードの詳細については、QuickSight ユーザーガイドのAmazon S3 ファイルを使用したデータセットの作成」を参照してください。 QuickSight

Amazon Redshift へのレポートデータのロード

このセクションでは、CUR AWS を Amazon Redshift にアップロードして AWS コストと使用状況を分析する方法について説明します。

> **Important**
> Amazon Redshift の列では、大文字と小文字を区別しません。また、ユーザー定義タグよりも厳格な文字制限があります。Amazon Redshift とユーザー定義タグ間の競合を防ぐため、はタグをタグ `userTag0`、`userTag1`、 `userTag2`など AWS に置き換えます。Amazon Redshift テーブルを作成してレポートをアップロードしたら、 AWS定義済みタグをユーザー定義タグにマッピングする Amazon Redshift テーブルを作成できます。タグテーブルを使って、元のタグを検索することができます。

例えば、OWNER や Owner というタグがある場合、「owner」という名前の 2 つの列を使用してテーブルを作成することを Amazon Redshift は許可しません。代わりに、OWNER と Owner ではなく、userTag0 と userTag1 という列を持つレポートテーブルを作成し、次に remappedUserTag と userTag という列を持つテーブルを作成します。remappedUserTag 列には AWS定義されたタグ userTag0 と が格納されuserTag1、userTag列には元のタグが格納OWNERされます。 Owner

AWS には、Amazon Redshift テーブルの作成、レポートのアップロード、タグテーブルの作成、すべてのタグ行のタグテーブルへの挿入を行うためのコマンドが用意されています。コマンドは、S3 のマニフェストRedshiftCommands.sqlファイルとともに保存されている ファイルと、請求情報とコスト管理コンソールの Redshift ファイルヘルパーファイルで提供されます。 には、RedshiftCommand ファイルのアップロードでコマンドを報告するための RedshiftManifest ファイル AWS も用意されています。 RedshiftCommand RedshiftManifest ファイルを削除すると、RedshiftCommands ファイルのコピーコマンドで問題が生じます。

請求およびコストマネジメントコンソールで **RedshiftCommands.sql** ファイルを見つけるには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. Amazon Redshift にアップロードするレポートを選択します。
4. [以下のサービスでのレポート表示が有効:]の横の [Amazon Redshift] を選択します。
5. ダイアログボックスからコマンドをコピーし、SQL クライアントに貼り付けます。

次の手順では、データベースや Amazon Redshift に詳しいことを前提としています。

コストと使用状況レポートを Amazon Redshift にアップロードするには

1. Amazon Redshift クラスターを作成します。

   詳細については、「Amazon Redshift 管理ガイド」の「クラスターの作成」を参照してください。
2. にサインイン AWS Management Console し、https://console.aws.amazon.com/s3/ で Amazon S3 コンソールを開きます。
3. CUR を保存する Amazon S3 AWS の場所に移動します。

4. `RedshiftCommands.sql` ファイルを開きます。

   ファイルには、Amazon Redshift テーブルを作成し、Amazon S3 から AWS CUR をアップロードし、ユーザー定義タグを Amazon Redshift にインポートできるようにするタグテーブルを作成するためのカスタマイズされたコマンドが含まれています。

5. copy コマンドで、*<AWS_ROLE>* を、CUR を保存する Amazon S3 バケットにアクセスする権限を持つ IAM ロールの ARN AWS に置き換えます。

6. *<S3_BUCKET_REGION>* を、Amazon S3 バケットがあるリージョンに置き換えます。例えば、`us-east-1`。

7. SQL クライアントを使用してクラスターに接続します。

   詳細については、Amazon Redshift管理ガイドの Amazon Redshift クラスターとデータベースへのアクセスを参照してください。

8. SQL コマンドを `RedshiftCommands.sql` ファイルから SQL クライアントに、次の順序でコピーします:

   - CREATE TABLE - このコマンドは、レポートに一致するようにカスタマイズされたスキーマを使って、Amazon Redshift テーブルを作成します。
   - copy - このコマンドは、提供された IAM ロールを使用して S3 から Amazon Redshift に AWS CUR ファイルをアップロードします。
   - CREATE TAG TABLE - このコマンドは、 AWS定義タグをユーザー定義タグにマッピングできるようにする、テーブルを作成します。
   - INSERT - このコマンドは、ユーザー定義タグをタグテーブルに挿入します。

9. CUR から Amazon Redshift AWS にすべてのデータをコピーしたら、SQL を使用してデータをクエリできます。Amazon Redshift でのデータのクエリの詳細については、Amazon Redshift データベースデベロッパーガイドの Amazon Redshift SQL を参照してください。

> **Note**
>
> コストと使用状況レポートの列数は、新しいコスト配分タグの作成やサービスでの新しい製品属性の追加などに伴って、毎月変わる場合があります。毎月 CUR AWS から新しいテーブルにデータをコピーし、関心のある列をmonth-by-monthテーブルにコピーすることをお勧めします。

Billing Conductor を使用したコストと使用状況レポートの設定

Billing Conductor で作成する請求グループごとに、見積り AWS コストと使用状況レポート (AWS CUR) を作成できます。見積り CUR AWS のファイル形式、粒度、列は、標準の CUR AWS と同じです。プロフォーマには、所定の期間に使用可能な最も包括的なコストと使用状況のデータが含まれています。詳細については、「Billing Conductor ユーザーガイド」を参照してください。

トピック

- Billing Conductor CUR AWS と標準 CUR AWS の違いを理解する
- 請求グループのプロフォーマコストと使用状況レポートを作成します

Billing Conductor CUR AWS と標準 CUR AWS の違いを理解する

Billing Conductor 設定を使用して作成された標準のコストと使用状況レポートと見積り AWS CUR にはいくつかの違いがあります。

- 標準の CUR AWS は、一括請求ファミリーの各アカウントのコストと使用量を計算します。請求グループあたりの見積り CUR AWS には、計算時に請求グループのアカウントのみが含まれます。
- 標準の AWS CUR は、請求書列を一度入力すると、請求書が によって生成されます AWS。見積り AWS CUR は請求書列に入力されません。現在、見積り請求データ AWS に基づいて が生成または発行する請求書はありません。

請求グループのプロフォーマコストと使用状況レポートを作成します

請求グループの見積り CUR AWS を生成するには、次の手順に従います。

請求グループのプロフォーマコストと使用料レポートを作成するには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. [レポートテーブル] の右上にある [設定] を選択します。
4. プロフォーマデータビューをオンにしてください。
5. [有効化] を選択します。

6. [レポートを作成] を選択します。
7. [レポート名] に、レポートの名前を入力します。
8. [データビュー] で、[プロフォーマ] を選択します。
9. 請求グループを選択してください。
10. [その他のレポート詳細] で、[リソース ID を含める] を選択して各リソースの ID をレポートに含めます。
11. データ更新設定で、 が請求書の確定後にアカウントに返金、クレジット、またはサポート料金 AWS を適用する場合に、コストと使用状況レポートを更新するかどうかを選択します。レポートが更新されると、新しいレポートが Amazon S3 にアップロードされます。
12. [次へ] を選択します。
13. [S3 バケット] で、[設定] を選択します。
14. [S3 バケットの設定] ダイアログボックスで、次のいずれかを実行します。
    - ドロップダウンリストから既存のバケットを選択し、[次へ] を選択します。
    - バケット名と新しいバケットを作成する AWS リージョンを入力し、次へを選択します。
15. バケットポリシーを確認し、[このポリシーが正しいことを確認しました] を選択した後、[保存] を選択します。
16. [レポートパスのプレフィックス)] に、レポート名に付加するレポートパスのプレフィックスを入力します。

    このステップは Amazon Redshift または QuickSight ではオプションですが、Amazon Athena では必須です。プレフィックスを指定しない場合、既定のプレフィックスは、ステップ 7 でレポートに指定した名前とレポートの日付範囲です。形式は次のとおりです。`/report-name/date-range/`
17. [時間粒度] で、次のいずれかを選択します。
    - 時間単位: レポートの明細項目を 1 時間ごとに集計する場合に選択します。
    - 日単位: レポートの明細項目を 1 日ごとに集計する場合に選択します。
18. [レポートバージョニング] で、レポートの各バージョンでレポートの以前のバージョンを上書きするのか、以前のバージョンに加えて配信するのかを選択します。

    レポートを上書きすると、Amazon S3 ストレージコストを節約できます。新しいレポートバージョンを配信することで、時間の経過とともに請求データの監査性を向上させることができます。

19. レポートデータ統合では、コストと使用状況レポートを Amazon Athena、Amazon Redshift、または QuickSight にアップロードするかどうかを選択します。レポートは、以下の形式で圧縮されています。
    - Athena: parquet 形式
    - Amazon Redshift または QuickSight: .gz 圧縮
20. [次へ] を選択します。
21. レポートの設定を確認したら、[確認して完了] を選択します。

データディクショナリ

コストと使用状況レポートには、使用状況に関する詳細が含まれています。以下のセクションでは、レポートに表示される列のサブセットおよび対応する定義を一覧して説明します。

AWS コストと使用状況レポート (AWS CUR) に表示される列の完全なリストと、列が適用されるサービスをダウンロードするには、Column_Attribute_Service.zip をダウンロードします。このカンマ区切り値 (CSV)リストには、Identity, Bill, LineItem, Reservation, Pricing, Product 列が含まれます。

すべてのコストと使用状況レポートには、Identity, Bill, LineItem 列が含まれています。他のすべての列は、毎月の AWS 使用量がそれらの列に入力するデータを生成する場合にのみレポートに含まれます。

トピック

- ID の詳細
- 請求の詳細
- 明細項目の詳細
- 予約の詳細
- 料金詳細
- 製品の詳細
- リソースタグの詳細
- Savings Plans の詳細
- Cost Categoriesの 詳細
- ディスカウントの詳細

- 分割明細項目の詳細

ID の詳細

AWS コストと使用状況レポートの identityヘッダーの列は、すべてのコストと使用状況レポートに表示される静的フィールドです。

レポート内の ID 明細項目を使用して、複数の AWS CUR ファイル間で分割された特定の明細項目を検索できます。これらには次の列が含まれます。

identity/LineItemId

- 説明: このフィールドは明細項目ごとに生成され、特定のパーティション内で一意です。これは、CUR の配信全体 (つまり、更新内のすべてのパーティション) AWS でフィールドが一意であることを保証するものではありません。明細項目 ID は、異なるコストと使用状況レポート間で一貫性がないため、異なるレポート間の同じ明細項目を識別するために使用することはできません。
- 例: 11 月 29 日分として作成されるレポートは大きいため、複数のファイルが必要になる可能性があります。LineItemId は 11 月 29 日の AWS CUR ファイル間では一貫していますが、11 月 30 日のレポートの同じリソースの LineItemId とは一致しません。

identity/TimeInterval

- 説明: この明細項目が適用される時間間隔、形式は `YYYY-MM-DDTHH:mm:ssZ/YYYY-MM-DDTHH:mm:ssZ` となります。時間の間隔は UTC で表され、レポートの詳細度に応じて毎日または毎時のいずれかになります。
- 例: TimeInterval `2017-11-01T00:00:00Z/2017-12-01T00:00:00Z` には、2017 年の 11 月全体が含まれます。

請求の詳細

AWS コストと使用状況レポートの billヘッダーの列は、すべてのコストと使用状況レポートに表示される静的フィールドです。レポートの請求明細項目を使用して、請求タイプや請求期間の開始/終了など、レポートの対象となる特定の請求に関する詳細を確認できます。これには以下の列が含まれます。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

B

bill/BillingEntity

請求書またはトランザクションが他の AWS サービスの購入用 AWS Marketplace か購入用かを識別するのに役立ちます。可能な値は以下のとおりです:

- AWS - AWS Marketplaceでのトランザクションはなく、 AWS サービスに関するトランザクションを特定します。
- AWS Marketplace - AWS Marketplaceでの購入を特定します。

bill/BillingPeriodEndDate

このレポートの対象となる請求期間の終了日 (UTC)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。

bill/BillingPeriodStartDate

このレポートの対象となる請求期間の開始日 (UTC)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。

bill/BillType

このレポートの対象となる請求書の種類。請求書の種類は 3 つあります。

- 応当日 – 当月中に使用したサービスの明細項目
- 購入 – 前払いサービス料金の明細項目
- 返金 – 返金の明細項目

I

bill/InvoiceId

特定の明細項目に関連付けられた ID。レポートが確定するまでは、[InvoiceId] が空白になります。

bill/InvoicingEntity

請求書を発行する AWS エンティティ。可能な値は以下のとおりです:

- Amazon Web Services, Inc. – 世界中のお客様に請求書を発行する事業体 (該当する場合)。
- Amazon Web Services India Private Limited – インドを拠点とするお客様に請求書を発行する事業体。

- Amazon Web Services South Africa Proprietary Limited – 南アフリカのお客様に請求書を発行する事業体。

P

bill/PayerAccountId

支払いアカウントのアカウント ID。の組織の場合 AWS Organizations、これは管理アカウントのアカウント ID です。

明細項目の詳細

AWS コストと使用状況レポートの lineItemヘッダーの列は、すべてのコストと使用状況レポートに表示される静的フィールドです。これには、あなたの使用に伴うすべてのコストと使用状況の情報が含まれます。これには以下の列が含まれます。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

A

lineItem/AvailabilityZone

この明細項目をホストするアベイラビリティーゾーン。例えば、`us-east-1a`、`us-east-1b` です。

B

lineItem/BlendedCost

`BlendedRate` に `UsageAmount` を乗じた値。

> **Note**
>
> [BlendedCost] は、[LineItemType] が [Discount] である明細項目については空白になります。ディスカウントは、メンバーアカウントと SKU で集計されたメンバーアカウントの非ブレンドコストのみを使用して計算されます。そのため、[BlendedCost] はディスカウントに使用できません。

lineItem/BlendedRate

`BlendedRate` は、組織全体で各 SKU に発生した平均コストです。

例えば、Amazon S3 ブレンドレートは、ストレージの合計コストを、1 か月間に保存されたデータ量で割ったものです。RI を持つアカウントの場合、ブレンドレートは RI とオンデマンドインスタンスの平均コストとして計算されます。

ブレンドレートは管理アカウントレベルで計算され、各メンバーアカウントにコストを割り当てるために使用されます。詳細については、AWS Billing ユーザーガイドのブレンドレートとコストを参照してください。

C

lineItem/CurrencyCode

この明細項目の通貨。すべての AWS お客様は、デフォルトで米ドルで請求されます。請求用の通貨を変更するには、AWS Billing ユーザーガイドの請求の支払いに使用する通貨の変更を参照してください。

L

lineItem/LegalEntity

特定の製品またはサービスの登録販売者。ほとんどの場合、請求書を発行する事業体と法律上の事業体は同じです。サードパーティーの AWS Marketplace トランザクションでは値が異なる場合があります。可能な値は以下のとおりです:

- Amazon Web Services, Inc. – AWS サービスを販売するエンティティ。
- Amazon Internet Services India Private Limited – インドでの AWS サービスのリセラーとして機能する現地インド法人。

lineItem/LineItemDescription

明細項目タイプの説明。たとえば、使用料の明細項目は、特定の期間に発生した使用タイプを要約したものです。

サイズ柔軟な RI の場合、説明は、利点が適用された RI に対応します。たとえば、明細項目が `t2.micro` に対応し、`t2.small` RI がその使用に適用された場合、lineItem/LineItemDescription には `t2.small` と表示されます。

RI ディスカウントが適用された使用料の明細項目の説明には、明細項目の対象となる料金プランが含まれています。

lineItem/LineItemType

この明細項目の対象となる請求の種類。以下の種類が対象となります。

- BundledDiscount – 別のサービスまたは機能の使用に基づいて、サービスまたは機能の無料または割引された使用を提供する使用ベースの割引。
- Credit – 請求書 AWS に適用されたクレジット。詳細については、[説明] 列を参照してください。AWS が請求額の確定後に当月のクレジットをアカウントに適用した場合、 AWS は確定後のレポートを更新することがあります。
- Discount – の使用 AWS に適用された割引。この特定の項目名は異なる場合があり、ディスカウントに基づいて解析する必要があります。詳細については、lineItem/LineItemDescription列を参照してください。
- DiscountedUsage – リザーブドインスタンス (RI) の利点があったインスタンスのレート。
- Fee – サブスクリプションに対して支払った前払いの年額料金。たとえば、[全前払い RI] または [一部前払い RI)] として支払った前払い金が該当します。
- Refund – が AWS 返金した金額のマイナスの料金。が請求書を確定した翌月の返金をアカウント AWS に適用する場合は、確定後にレポートを更新する AWS 可能性があるので、説明列を確認してください。
- RIFee – サブスクリプションの月額定期料金。たとえば、毎月支払う [一部前払い RI]、[前払いなし RI]、[全額前払い] の定期的な料金を指します。RIFee は全額前払い予約が 0 USD になる場合がありますが、reservation/AmortizedUpfrontFeeForBillingPeriodやreservation/ReservationARNなどの他の列を提供するために、この行はこれらの予約タイプに対しても事前入力されます。
- Tax – 請求書 AWS に適用されたすべての税金。VAT、米国の消費税などが該当します。
- Usage – オンデマンドインスタンス料金で課金されるすべての使用量。
- SavingsPlanUpfrontFee – 全額前払いまたは一部前払いの Savings Plans の購入による 1 回限りの前払い料金。
- SavingsPlanRecurringFee – 前払いなしまたは一部前払いSavings Plan に対応する定期的な時間単位の料金。Savings Plan 定期料金は、前払いなしまたは一部前払い Savings Plan を購入した日に、最初に請求書に追加されます。初回購入後、 は各請求期間の最初の日に定期的な料金 AWS を追加します。
- SavingsPlanCoveredUsage – Savings Plans の対象となるオンデマンドコスト。Savings Plan の対象となる使用量明細項目は、対応する Savings Plan 否定項目によって相殺されます。

- SavingsPlanNegation – 対応する Savings Plan s対象使用量項目に関連付けられている Savings Plan s特典によるオフセットコスト。

Savings Plans 明細項目の詳細と例については、Savings Plans について を参照してください。

N

lineItem/NetUnblendedCost

品目に対して支払っている実際のディスカウント後の費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

lineItem/NetUnblendedRate

その明細項目に対して支払っている実ディスカウント後レート。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

lineItem/NormalizationFactor

インスタンスがテナンシーを共有している限り、 はすべてのリージョン Linux または Unix Amazon EC2 および Amazon RDS RI 割引をインスタンスファミリーと AWS リージョンのすべてのインスタンスサイズに適用 AWS できます。これは、組織内のメンバーアカウントの RI ディスカウントにも適用されます。Amazon EC2 および Amazon RDS のすべての新規および既存のサイズ柔軟な RI のサイズは、インスタンスサイズに基づいた正規化係数により決定されます。次の表は、各インスタンスサイズ AWS に適用される正規化係数を示しています。

Amazon EC2 サイズ柔軟な RI の正規化係数

インスタンスサイズ	正規化係数
nano	0.25
micro	0.5
small	1
medium	2

インスタンスサイズ	正規化係数
`large`	4
`xlarge`	8
`2xlarge`	16
`4xlarge`	32
`8xlarge`	64
`10xlarge`	80
`16xlarge`	128
`32xlarge`	256

lineItem/NormalizedUsageAmount

サイズ柔軟な RI のために発生した使用量。正規化単位。[NormalizedUsageAmount] は、[UsageAmount] に [NormalizationFactor] を乗算したものに等しくなります。

O

lineItem/Operation

この明細項目の対象となる特定の AWS オペレーション。これは、明細項目の特定の使用状況を表します。たとえば、`RunInstances` の値は Amazon EC2 インスタンスのオペレーションを示します。

P

lineItem/ProductCode

測定された製品のコード。例えば、Amazon EC2 は、Amazon Elastic Compute Cloud の製品コードです。

R

lineItem/ResourceId

(オプション) レポートに個々のリソース ID を含めることを選択した場合、この列にはプロビジョンしたリソースの ID が表示されます。例えば、Amazon S3 ストレージバケット、Amazon EC2 コンピューティングインスタンス、または Amazon RDS データベースは、すべてリソース ID を持つことができます。このフィールドは、インスタンス化されたホストに関連付けられていない使用タイプ (データ転送、API リクエストなど) と明細項目タイプ (ディスカウント、クレジット、税金など) では空白です。次の表は、一般的な AWS サービスのリソース識別子のリストを示しています。

AWS リソース識別子

AWS サービス	リソース識別子
Amazon CloudFront	ディストリビューション ID
Amazon CloudSearch	検索ドメイン
Amazon DynamoDB	DynamoDB テーブル
Amazon Elastic Compute Cloud - Amazon EBS	Amazon EBS ボリューム
Amazon Elastic Compute Cloud	[インスタンス ID]
Amazon Elastic Compute Cloud - CloudWatch	インスタンス ID の CloudWatch の料金
Amazon EMR	MapReduce クラスター
Amazon ElastiCache	キャッシュクラスター
Amazon OpenSearch Service	検索ドメイン
Amazon S3 Glacier	ボールト

AWS サービス	リソース識別子
Amazon Relational Database Service	データベース
Amazon Redshift	Amazon Redshift クラスター
Amazon Simple Storage Service	Amazon S3 バケット
Amazon Virtual Private Cloud	VPN ID
AWS Lambda	Lambda 関数の名前

T

lineItem/TaxType

この明細項目 AWS に適用された税金のタイプ。

U

lineItem/UnblendedCost

`UnblendedCost` は、`UnblendedRate` に `UsageAmount` を乗じた値です。

lineItem/UnblendedRate

を使用するアカウントの一括請求では AWS Organizations、非ブレンドレートは、個々のアカウントのサービス使用量に関連付けられたレートです。

RI ディスカウントが適用されている Amazon EC2 と Amazon RDS 明細項目の場合、`UnblendedRate` は 0 です。RI ディスカウントが適用されている明細項目は、`LineItemType` が `DiscountedUsage` です。

lineItem/UsageAccountId

この明細項目を使用したアカウントの ID。組織の場合、これは管理アカウントまたはメンバーアカウントのいずれかになります。このフィールドを使用して、アカウント別のコストや使用状況を追跡することができます。

lineItem/UsageAmount

指定した期間に発生した使用量。サイズ柔軟なリザーブドインスタンスについては、代わりに reservation/TotalReservedUnits 列を使用します。

> **Note**
> 特定のサブスクリプション料金では、UsageAmount が `0` になります。

lineItem/UsageEndDate

対応する明細項目の終了日時は UTC で表されます (その時刻は含まない)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。

lineItem/UsageStartDate

明細項目の開始日時は UTC で表されます (その時刻を含む)。形式は `YYYY-MM-DDTHH:mm:ssZ` です。

lineItem/UsageType

この明細項目の使用状況の詳細。例えば、`USW2-BoxUsage:m2.2xlarge` は米国西部 (オレゴン) リージョンの M2 ハイメモリダブルエクストララージインスタンスについて説明します。

予約の詳細

AWS コストと使用状況レポートの reservationヘッダーの列には、リザーブドリソースの詳細が表示されます。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

A

reservation/AmortizedUpfrontCostForUsage

- 説明: 全額前払い RI および一部前払い RI の最初の前払い料金は、使用時間に応じて償却されます。この値は `RIAmortizedUpfrontFeeForBillingPeriod` * `The normalized usage amount for DiscountedUsage line items` / `The normalized usage amount for the RIFee` と等しくなります。前払いなし RI には前払いはないため、前払いなし RI の値は `0` です。現時点では、Dedicated Hosts 予約にはこの値を提供していません。将来のアップデートで変更を行う予定です。
- 該当する明細項目: DiscountedUsage
- サンプル値： `0.05`, `0.17`, `0.15`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

reservation/AmortizedUpfrontFeeForBillingPeriod

- 説明: この予約の前払い料金のうち、請求期間中の費用がどれくらいになるかを示します。全額前払い RI および一部前払い RI の最初の前払い料金は、今月償却されます。前払いなし RI には前払い料金はないため、前払いなし RI の値は `0` です。現時点では、Dedicated Hosts 予約にはこの値を提供していません。将来のアップデートで変更を行う予定です。
- 該当する明細項目: RIFee
- サンプル値： `29.15`, `200.67`, `214.43`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache

- Amazon RDS

reservation/AvailabilityZone

- 説明: この明細項目に関連付けられたリソースのアベイラビリティーゾーン。
- 該当する明細項目: Fee、Refund、RIFee
- サンプル値： `us-east-1`, `us-east-1b`, `eu-west-1b`, `ap-southeast-2a`
- サービス:
  - Amazon EC2

E

reservation/EffectiveCost

- 説明: 実質的時間料金に平均化された、RI の前払い料金と時間料金の両方の合計。`EffectiveCost` は、`amortizedUpfrontCostForUsage` を `recurringFeeForUsage` に加算することにより計算されます。詳細については、Amazon EC2 リザーブドインスタンスの料金を参照してください。
- 該当する明細項目: DiscountedUsage
- サンプル値： `0.23`, `0.68`, `0.10`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon ElastiCache
  - Amazon RDS

reservation/EndTime

- 説明: 関連付けられた RI リース期間の終了日。
- 該当する明細項目: RIFee
- サンプル値： `2019-05-15T04:23:14.000Z`, `2020-02-08T17:32:15.000Z`, `2019-07-14T00:00:33.000Z`
- サービス:

- Amazon EC2
- OpenSearch Service
- Amazon Redshift
- Amazon ElastiCache
- Amazon RDS

M

reservation/ModificationStatus

- 説明: RI リースが変更されたかどうかを示します。
  - Original (オリジナル): 購入した RI は 1 回も変更されていません。
  - System (システム): 購入した RI は、コンソールまたは API を使用して変更されました。
  - 手動： 購入した RI が AWS サポート サポートを使用して変更されました。
  - ManualWithData: 購入した RI が AWS サポート 支援を使用して変更され、RI の見積りが AWS 計算されました。
- 該当する明細項目: RIFee
- サンプル値： `Original`, `System`, `Manual`, `ManualWithData`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

N

reservation/NetAmortizedUpfrontCostForUsage

全前払い RI および一部前払い RI の最初の前払い料金は、該当する場合、使用時間に応じて償却されます。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetAmortizedUpfrontFeeForBillingPeriod

請求期間中の予約の前払い料金の費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetEffectiveCost

RI の前払い料金と時間単価の合計を、平均して実効時間単価とします。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetRecurringFeeForUsage

継続利用料金のディスカウント後の金額。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetUnusedAmortizedUpfrontFeeForBillingPeriod

請求期間中に償却された正味の未使用の前払い料金。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetUnusedRecurringFee

一部前払い RI と前払いなし RI の未使用の予約時間に関連する定期的な料金。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NetUpfrontValue

ディスカウントが適用された RI の前払い額。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

reservation/NormalizedUnitsPerReservation

- 説明: 予約サブスクリプションの各インスタンスの正規化された単位の数。
- 該当する明細項目: RIFee
- サンプル値： `1316, 54.5, 319`
- サービス:
  - Amazon RDS

reservation/NumberOfReservations

- 説明: このサブスクリプションに含まれる予約の数。たとえば、1 件のサブスクリプションには 4 件の RI 予約を関連付けることができます。

- 該当する明細項目: Fee、RIFee、Refund、Credit
- サンプル値： `5, 50, 500`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

R

reservation/RecurringFeeForUsage

- 説明: 一部前払い RI および前払いなし RI の場合、使用時間に応じて償却される定期的料金です。この値は `The unblended cost of the RIFee * The sum of the normalized usage amount of Usage line items / The normalized usage amount of the RIFee for size flexible Reserved Instances` と等しくなります。全額前払い RI には `0` より大きい定期料金支払いがないため、全額前払いRI の値は `0` です。
- 該当する明細項目: DiscountedUsage
- サンプル値： `0.139, 0.729, 0.018`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon ElastiCache
  - Amazon RDS

reservation/ReservationARN

- 説明: この明細項目が活用した RI の Amazon リソースネーム (ARN)。これは「RI リース ID」とも呼ばれます。これは、この特定の AWS リザーブドインスタンスの一意の識別子です。この値の文字列には、 AWS サービス名と、RI を購入したリージョンも含まれています。

- 該当する明細項目: Fee、RIFee、DiscountedUsage、Refund、Credit
- サンプル値： `arn:aws:ec2:us-east-1:074108124787:reserved-instances/1d3fbc13-f181-4c40-9dd6-12b345678de9`, `arn:aws:ec2:us-east-1:499958231354:reserved-instances/be41234c3-b5c0-403e-a80c-1cfd12345678`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

S

reservation/StartTime

- 説明: 関連付けられたリザーブドインスタンスの期間の開始日。
- 該当する明細項目: RIFee
- サンプル値： `2018-07-29T02:56:10.000Z`, `2017-08-21T15:58:47.000Z`, `2019-02-01T22:01:34.000Z`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

reservation/SubscriptionId

- 説明: 明細項目を関連オファーとマッピングする一意の識別子。RI ARN を AWS リザーブドインスタンスの識別子として使用することをお勧めしますが、両方を使用できます。
- 該当する明細項目: Fee、RIFee、DiscountedUsage、Refund、Credit、および Usage

- サンプル値： `123456789, 111122222, 333344444`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

T

reservation/TotalReservedNormalizedUnits

- 説明： 予約サブスクリプションのすべてのインスタンスの予約済み正規化単位の合計数。 は、`reservation/NormalizedUnitsPerReservation`に を乗算して正規化単位の合計を AWS 計算します`reservation/NumberOfReservations`。
- 該当する明細項目: DiscountedUsage
- サンプル値： `40320, 3647.99, 17928.77`
- サービス:
  - Amazon EC2
  - Amazon RDS

reservation/TotalReservedUnits

- 説明: `TotalReservedUnits` により、Fee および RIFee の両方の明細項目に個別の値が入力されます。
  - Fee 明細項目: 期間全体に対してサブスクリプションで購入した合計リース数量に対する、予約されたユニットの合計数。

    これは、`NumberOfReservations` に `UnitsPerReservation` を掛けて計算されています。例: 5 RI x 744 時間/月 x 12 か月 = 44,640。
  - RIFee 明細項目 (毎月の定期的な料金): 特定の RI サブスクリプションでの Amazon EC2 時間の合計など、サブスクリプションで利用できる合計ユニット数。

    例: 5 RI x 744 時間 = 3,720。

- 該当する明細項目: Fee、RIFee、Refund、Credit
- サンプル値： `26208`, `98.19`, `15796`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

U

reservation/UnitsPerReservation

- 説明: `UnitsPerReservation` により、Fee および RIFee の両方の明細項目に個別の値が入力されます。
  - Fee 明細項目: サブスクリプション期間に対して購入した RI 時間の合計数など、サブスクリプションに予約した合計ユニット数。

    例: 744 時間/月 x 12 か月 = 8,928 合計時間/ユニット。
  - RIFee 明細項目 (毎月の定期的な料金): 特定の RI サブスクリプションでの Amazon EC2 時間の合計など、サブスクリプションで利用できる合計ユニット数。

    例: 1 ユニット x 744 時間 = 744。
- 該当する明細項目: Fee、RIFee、Refund、Credit
- サンプル値： `334.0`, `486.72`, `18455`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon DynamoDB
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

reservation/UnusedAmortizedUpfrontFeeForBillingPeriod

- 説明: 全額前払い RI および一部前払い RI の最初の前払い料金の amortized-upfront-fee-for-billing-period-column 償却部分。前払いなし RI には前払いはないため、前払いなし RI の値は `0` です。現時点では、Dedicated Hosts 予約にはこの値を提供していません。将来のアップデートで変更を行う予定です。
- 該当する明細項目: RIFee
- サンプル値： `6.05`, `1.97`, `0.17`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

reservation/UnusedNormalizedUnitQuantity

- 説明: この請求期間中に使用しなかった、サイズ柔軟なリージョン RI の未使用の正規化されたユニットの数。
- 該当する明細項目: RIFee
- サンプル値： `25.00`, `3.50`, `274.33`
- サービス:
  - Amazon RDS

reservation/UnusedQuantity

- 説明: この請求期間中に使用しなかった RI 時間数。
- 該当する明細項目: RIFee 明細項目
- サンプル値： `209.65110408`, `191.00000000`, `176.00000000`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon Redshift

- Amazon ElastiCache
- Amazon RDS

reservation/UnusedRecurringFee

- 説明: 一部前払い RI と前払いなし RI の未使用の予約時間に関連する定期的な料金。全額前払い RI には `0` より大きい定期料金がないため、全額前払い RI の値は `0` です。
- 該当する明細項目: RIFee
- サンプル値： `0.02971114`, `0.19190000`, `1.37280000`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

reservation/UpfrontValue

- 説明： AWS リザーブドインスタンスに支払われた前払い料金。前払いのない RI の場合、この値は `0` になります。
- 該当する明細項目: RIFee
- サンプル値： `150.00`, `1000.00`, `2000.00`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon Redshift
  - Amazon ElastiCache
  - Amazon RDS

料金詳細

AWS コストと使用状況レポートの pricingヘッダーの列には、明細項目の料金が含まれています。料金列は AWS Price List Service API に基づいています。 AWS Price List Service API には、無料

利用枠料金、スポットインスタンス、 の製品、前払い年間サブスクリプション料金 (Fee) AWS Marketplace、および月額定期料金 () は含まれませんRIFee。予約列には次のようなものがあります。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

L

pricing/LeaseContractLength

RI の予約期間。

O

pricing/OfferingClass

- 説明: リザーブドインスタンスの提供クラスについて説明します。リザーブドインスタンスを購入するとき、スタンダードあるいはコンバーティブルの提供クラスから選択できます。
- サンプル値： `Standard`, `Convertible`
- サービス:
  - Amazon DynamoDB
  - Amazon EC2
  - Amazon ElastiCache
  - OpenSearch Service
  - Amazon RDS
  - Amazon Redshift

P

pricing/publicOnDemandCost

パブリックオンデマンドインスタンスレートに基づく明細項目の合計コスト。複数のオンデマンドパブリックコストが発生する SKU がある場合は、最も高い階層の同等のコストが表示されます。無料利用枠や階層型価格を提供するサービスなどが該当します。

pricing/publicOnDemandRate

特定の使用料の明細項目に対する、この請求期間のパブリックオンデマンドインスタンスレート。複数のオンデマンドパブリック料金が発生する SKU がある場合、最も高い階層の同等の料金が表示されます。無料利用枠や階層型価格を提供するサービスなどが該当します。

pricing/PurchaseOption

この明細項目に選択した支払い方法。有効な値は、`All Upfront`、`Partial Upfront`、`No Upfront` です。

R

pricing/RateCode

製品/オファー/価格-枠の組み合わせの一意のコード。製品および期間の組み合わせは、無料利用枠、低使用利用枠、高使用利用枠など、複数の価格ディメンションを持つことができます。

pricing/RateId

明細項目のレートの ID。

T

pricing/term

AWS 使用量がリザーブドかオンデマンドか。

U

pricing/unit

が使用コストの計算 AWS に使用した料金単位。例えば、Amazon EC2 インスタンスの使用料の料金単位は時間数です。

製品の詳細

product列には、費用が発生した製品に関するメタデータと明細項目が表示されます。製品列は動的であり、コストと使用状況レポートの表示は、請求期間中の製品の使用状況によって異なります。料金列は AWS Price List Service API に基づいています。 AWS Price List Service API には、無料利

用枠料金、スポットインスタンス、 AWS Marketplace の製品、前払い年間サブスクリプション料金 (`Fee`) 、月額定期料金 () は含まれません`RIFee`。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | XYZ

A

product/APICalls

- 説明: DevOps Guru サービスを有効にするために DevOps Guru サービスが使用する API の数について説明します。
- サンプル値： `100`, `500`, `10000`
- サービス:
  - Amazon DevOps Guru

product/attachmentType

- 説明: Transit Gateway またはクラウド WAN サービスへのアタッチメントのタイプについて説明します。
- サンプル値： `VPC`, `AWS Site-to-Site VPN`, `AWS DirectConnect`, `Connect`, `Transit Gateway`
- サービス:
  - Amazon Virtual Private Cloud
  - AWS クラウド WAN

product/availability

- 説明： さまざまな AWS ストレージオプションの可用性について説明します。
- サンプル値： `99.99%`, `99.5%`
- サービス:
  - Amazon S3 Glacier
  - Amazon S3
  - AWS Elemental MediaStore
  - AWS RoboMaker

C

product/cacheType

- 説明: 頻繁な読み取りデータのパフォーマンスを向上させるために、読み取り専用 SSD キャッシュの HDD ベースのファイル システムで顧客によって選択されたプロビジョニングについて説明します。

  例えば、RC20 はファイルシステムの HDD ストレージ容量の 20% に自動的にサイズ変更される読み取り専用 SSD キャッシュが存在することを示します。
- サンプル値： `RC20`, `N/A`
- サービス:
  - Amazon FSx

product/capacitystatus

- 説明: キャパシティー予約のステータスを表します。
- サンプル値： `UnusedCapacityReservation`, `AllocatedCapacityReservation`, `Used`
- サービス:
  - Amazon EC2

product/clockspeed

- 説明： AWS インスタンスの動作速度について説明します。
- サンプル値： `2.4 GHz`, `2.6 GHz`
- サービス:
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon MQ
  - Amazon Neptune
  - Amazon RDS
  - AWS データベース移行サービス

product/component

- 説明： SageMaker AI の機能にマッピングします。

  たとえば、ユーザーが SageMaker AI ノートブックを実行している場合、製品には Notebook のコンポーネント属性があります。ユーザーが推論用にモデルをデプロイしてホストしている場合、コンポーネント属性が Hosting の製品が表示されます。
- サンプル値： `Notebook`, `Hosting`
- サービス:
  - Amazon SageMaker AI

D

product/databaseedition

- 説明: さまざまな開発、デプロイメントシナリオ、特定のアプリケーション目的に適したデータベースソフトウェアについて説明します。
- サービス:
  - Amazon RDS

product/dataTransfer

- 説明： AWS データ転送は、データが から AWS インターネットに移動するか、それぞれのリージョンまたはアベイラビリティーゾーンの AWS インスタンス間で移動されるたびに発生します。リージョン間およびアベイラビリティーゾーン間のデータ転送には、ギガバイト単位で従量制のコストがかかります。
- サービス:
  - AWS Systems Manager

product/dedicatedEbsThroughput

- 説明: インスタンス間の専用スループットを表します (例: Amazon EC2 インスタンスおよび Amazon EBS ボリューム)。使用するインスタンスタイプに応じて 500 ~ 10,000 メガビット/秒 (Mbps) を選択できます。専用スループットにより、Amazon EBS I/O と、EC2 インスタンスからのその他のトラフィックとの競合が最小限に抑えられ、Amazon EBS ボリュームに最適なパフォーマンスを得られます。

- サンプル値： `200 Mbps`, `Upto 5000 Mbps`
- サービス:
  - Amazon EC2
  - Amazon Neptune
  - Amazon RDS

product/deploymentoption

- 説明: 環境のインフラストラクチャがある場所について説明します。 AWS クラウドのデプロイモデルは、`public`、`on-premise`、`hybrid` です。
- サンプル値： `Multi-AZ`, `Single-AZ`
- サービス:
  - Amazon MQ
  - Amazon Neptune
  - Amazon RDS

product/description

- 説明： 特定の AWS サービスの説明。
- サービス:
  - AWS CodePipeline
  - AWS Device Farm
  - AWS Elemental MediaConvert
  - AWS Elemental MediaStore

product/destinationCountryISOCode

- 説明: SMS の送信先の国の ISO 3166-1 alpha-2 コードについて説明します。

  参照として、https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 をご覧ください。
- サンプル値： `FR`, `CO`, `MA`, `KN`, `PL`, `LV`, `LA`, `GB`, `ID`, `KR`, `MY`, `BR`, `MM`, `CA`, `VN`, `BD`, `BJ`, `AU`, `HK`, `AM`, `CZ`, `UA`, `PH`, `TW`, `ES`, `DE`, `NG`, `FI`, `SG`, `TH`, `IL`, `TR`, `JP`, `IT`, `PR`, `RU`, `EE`
- サービス:

- Amazon Simple Notification Service

product/directconnectlocation

- 説明: 顧客から AWS へのプライベート専用ネットワーク接続が存在する場所を指定します。
- サンプル値： `Equinix DC1 - DC6`, `Equinix DC10 - DC11`, `Global Switch Singapore`
- サービス:
  - AWS Direct Connect

product/directorysize

- 説明: ディレクトリまたはフォルダーのメタ情報を保存するために使用されるディスク上の容量。
- サービス:
  - AWS Direct Connect

product/directorytype

- 説明: ディレクトリがファイルか別のディレクトリかを指定します。
- サービス:
  - AWS Direct Connect

product/directorytypedescription

- 説明: ディレクトリに付けられたわかりやすい名前。
- サービス:
  - AWS Direct Connect

product/disableactivationconfirmationemail

- 説明: サービスの有効化を確認するメール送信機能を有効または無効にします。

product/durability

- 説明: 一定の年のオブジェクトの耐久性を表します。

- サンプル値： `99.999999999%`, `N/A`, `99.99%`
- サービス:
  - Amazon S3 Glacier
  - Amazon S3
  - AWS Elemental MediaStore

E

product/ebsOptimized

- 説明: Amazon EC2 インスタンスで Amazon EBS が最適化されているかどうかを示します。
- サンプル値： `Yes`, `No`
- サービス:
  - Amazon EC2

product/ecu

- 説明: Amazon EC2 インスタンスの整数処理能力を相対的に測定する EC2 コンピュートユニット (ECU) を示します。
- サンプル値： `9`, `100`, `variable`
- サービス:
  - Amazon EC2
  - OpenSearch Service
  - Amazon GameLift Servers
  - Amazon Redshift

product/endpointtype

- 説明: デバイスの接続先のリモート接続の特徴について説明します。

  例えば、`REST` (表現型状態転送) エンドポイントなどです。`REST` API (または `RESTful` API) は、`REST` アーキテクチャスタイルの制約に準拠したアプリケーションプログラミングインターフェイスであり、`RESTful` Web サービスとやり取りできます。
- サンプル値： `Ipsec`, `Amazon SQS`, `AWS Lambda`

- サービス:
  - Amazon SNS
  - Amazon VPC
  - Storage Gateway
  - S3 Glacier

product/enhancedNetworkingSupported

- 説明: インスタンスが拡張ネットワーキングをサポートしているかどうかを表します。拡張ネットワーキングでは、シングルルート I/O 仮想化 (SR-IOV) を使用して、サポートされるインスタンスタイプにおける高性能ネットワーキング機能が提供されます。
- サンプル値： `Yes`, No
- サービス:
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon Neptune
  - Amazon RDS
  - AWS データベース移行サービス

F

product/filesystemtype

- 説明: ローカルまたはリモートのストレージデバイスの詳細と、オペレーティングシステムの仕様について説明します。

product/findingGroup

- 説明: Security Hub に保存されている検出結果を有料にするか無料にするかを指定します。無料の場合は、理由も指定できます。
- サンプル値： `FreeFindingsIngestion-CrossRegion`, `FreeFindingsIngestion-FreeTier`, `FreeFindingsIngestion-FreeTrial`, `PaidFindingsIngestion`
- サービス:
  - AWS Security Hub

product/findingSource

- 説明: 検出結果が Security Hub コントロールによって生成されたのか、他のパートナーセキュリティ製品によって生成されたのかを指定します。
- サンプル値： `SecurityHubProduct`, `OtherProduct`
- サービス:
  - AWS Security Hub

product/freeUsageIncluded

- 説明: AWS 無料利用枠での無料利用は、すべてのリージョンで毎月計算され、自動的に請求額に適用されます。たとえば、使用するすべてのリージョンで 750 時間の Amazon EC2 Linux マイクロインスタンスを無料で利用できます。1 リージョンあたり 750 時間ではありません。
- サービス:
  - Amazon Inspector

product/fromLocation

- 説明: 使用が開始された場所を示します。
- サンプル値： `External`, `US East (N. Virginia)`, `Global`
- サービス:
  - Amazon CloudFront
  - AWS DataTransfer

product/fromRegionCode

- 説明： AWS サービスのソースリージョンコードについて説明します。詳細については、「product/regioncode」を参照してください。
- サンプル値： `ap-northeast-1`
- サービス:
  - Amazon RDS
  - Amazon EC2
  - Amazon VPC
  - AWS Direct Connect

product/fromLocationType

- 説明: 使用が開始された場所のタイプを示します。
- サンプル値： `AWS Region`, `AWS Edge Location`
- サービス:
  - AWS Direct Connect
  - AWS Elemental MediaConnect
  - Amazon CloudFront
  - Amazon Lightsail
  - AWS Shield

G

product/gpu

- 説明: GPUの数を表します。
- サンプル値： `16`, `32`
- サービス:
  - Amazon SageMaker AI
  - Amazon EC2

product/gpuMemory

- 説明: GPU メモリの詳細を表します。
- サンプル値： `16`, `32`
- サービス:
  - Amazon SageMaker AI
  - Amazon EC2

product/group

- 説明: 定義上類似している、またはグループ化された複数の製品の構成。たとえば、Amazon EC2 チームは製品を共有インスタンス、専用ホスト、専用用途に分類できます。
- サービス:

- AWS Certificate Manager
- AWS CodeCommit
- AWS Glue
- AWS IoT Analytics
- AWS Lambda

product/groupdescription

- 説明: 製品グループに付けられた略称。
- サービス:
  - AWS 予算
  - AWS Certificate Manager
  - AWS Lambda
  - Amazon SQS

I

product/insightsType

- 説明: 生成された Insight イベントのタイプを示します。
- サンプル値： `APICallVolume`
- サービス:
  - CloudTrail

product/instance

- 説明: Amazon EC2 インスタンスは、 AWS インフラストラクチャ上でアプリケーションを実行するための Amazon Elastic Compute Cloud (Amazon EC2) 内の仮想サーバーです。が提供する AMI AWS、ユーザーコミュニティ、または から選択できます AWS Marketplace。
- サンプル値： `T3`
- サービス:
  - Amazon EC2

product/instanceFamily

- 説明: Amazon EC2 インスタンスファミリーを示します。Amazon EC2 には、10 種類の異なるインスタンスタイプに多数のオプションが用意されています。各タイプに 1 つ以上のサイズオプションがあり、さまざまなタイプのアプリケーションのために最適化された異なるインスタンスファミリーに分類されています。
- サンプル値： `General Purpose`, `Memory Optimized`, `Accelerated Computing`
- サービス:
  - Amazon EC2
  - Amazon RDS
  - OpenSearch Service
  - Amazon ElastiCache
  - Amazon EMR

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/instanceSize

- 説明: リソースのインスタンスサイズを示します。
- サンプル値： `2vCPU`, `4vCPU`, `8vCPU`, `16vCPU`
- サービス:
  - Amazon CodeCatalyst

product/instanceType

- 説明: インスタンスのタイプ、サイズ、およびファミリーを表します。これらは、インスタンスの CPU、ネットワーク、およびストレージ容量を定義します。
- サンプル値： `t2.small`, `m4.xlarge`, `t2.micro`, `m4.large`, `t2.large`
- サービス:
  - Amazon EC2
  - Amazon RDS
  - OpenSearch Service
  - Amazon ElastiCache

- Amazon EMR

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/instanceTypeFamily

- 詳細: 指定された使用に関連付けられているインスタンスファミリー。
- サンプル値： `t2`, `m4`, `m3`
- サービス:
  - Amazon DocumentDB
  - Amazon RDS

product/integratingApi

- 説明： Amazon API Gateway などのサービス AWS を使用した でのアプリケーション統合、または Amazon AppFlow を使用したノーコード統合。

product/integratingService

- 説明： でのアプリケーション統合 AWS は、マイクロサービス、分散システム、サーバーレスアプリケーション内の分離されたコンポーネント間の通信に使用される一連のサービスです。アーキテクチャ全体をリファクタリングする必要はありません。アプリケーションをどのような規模でも切り離すことで、変更の影響を軽減でき、更新が容易になり、新機能をより早くリリースできるようになります。

product/intelAvxAvailable

- 説明: プロセスにインテル Advanced Vector Extension 命令セットがあるかどうかを表します。
- サンプル値： `Yes`, No.
- サービス:
  - Amazon EC2

product/intelAvx2Available

- 説明: プロセスにインテル Advanced Vector Extension 命令セット 2 があるかどうかを表します。
- サンプル値： `Yes`, No
- サービス:
  - Amazon EC2

product/intelTurboAvailable

- 説明: コアで Intel Turbo Technology を使用して周波数を上げることができるかどうかを表します。
- サンプル値： `Yes`, No
- サービス:
  - Amazon EC2

product/invocation

- 説明: EventBridge スケジューラーが API またはサービスに対して行う呼び出しについて説明します。
- サンプル値： `Scheduled Invocation`
- サービス:
  - Amazon CloudWatch Events

L

product/licenseModel

- 説明: インスタンスのライセンスモデルを表します。
- Sample value: `license-included`, `bring-your-own-license`, `general-public-license`
- サービス:
  - Amazon AppStream
  - Amazon EC2
  - Amazon MQ
  - Amazon Neptune

- Amazon RDS

product/location

- 説明: Amazon S3 バケットが存在するリージョンを示します。
- サンプル値： `Asia Pacific (Mumbai)`, `Asia Pacific (Seoul)`, `Canada (Central)`, `EU (London)`, `US West (Oregon)`
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/locationType

- 説明: タスクのエンドポイントを示します。
- サンプル値： `AWS Region`, `AWS Edge Location`, `Other`
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/logsDestination

- 説明:`AWS::Logs::Destination`リソースは、CloudWatch Logs の送信先を指定します。送信先には物理リソース (Amazon Kinesis データ ストリームなど) が含まれており、そのリソースをログイベントのストリームにサブスクライブできます。
- サンプル値： `AWS Region`, `AWS Edge Location`, `Other`
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB

    詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

M

product/maxIopsBurstPerformance

- 説明: Amazon EBS ボリュームの最大 IOPS バーストパフォーマンスを示します。
- Sample value: `3000 IOPS for volumes <= 1TB`
- サービス:
  - Amazon EC2

product/maxIopsvolume

- 説明: Amazon EBS ボリュームの 1 秒あたりの最大入力/出力を示します。
- Sample value: `16,000 (maxiops for a General Purpose SSD (gp2))`
- サービス:
  - Amazon EC2

product/maxThroughputvolume

- 説明: Amazon EBS ボリュームの最大ネットワークスループットボリュームを示します。

- サンプル値： `500 MiB/s`, `250 MiB/s`, `1000 MiB/s`, `40 - 90 MB/sec`
- サービス:
  - Amazon EC2
  - Amazon SageMaker AI

product/memory

- 説明: コンピュータが迅速に応答するために必要な命令やデータを格納するプレースホルダー電子機器。コンピューターバイトはストレージユニットを示します。
- サービス:
  - AWS Database Migration Service
  - DynamoDB Accelerator
  - Amazon DocumentDB
  - Amazon EC2

product/messageCountfee

- 説明: メータリングの使用タイプを説明し、その使用量がメッセージ数なのか、請求された料金なのかを示します。
- サンプル値： `CarrierFeeCount`, `MessageFees`, `MessageCount`, `CarrierFees`
- サービス:
  - Amazon Simple Notification Service

product/messageType

- 説明: SMS メッセージの種類について説明します。SNS はアウトバウンド SMS のみをサポートしていることに注意してください。
- サンプル値： `OutboundSMS`
- サービス:
  - Amazon Simple Notification Service

N

product/networkPerformance

- 説明: Amazon EC2 インスタンスのネットワークスループットを示します。
- サンプル値： `moderate`, `high`, `up to 10 GB`
- サービス:
  - Amazon EC2
  - Amazon RDS
  - Amazon ElastiCache
  - Amazon SageMaker AI
  - AWS データベース移行サービス

    詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/normalizationSizeFactor

- 説明: インスタンスサイズの正規化係数を表します。
- サンプル値： `nano - 0.25`, `micro - 0.5`, `medium - 2`, `xlarge - 8`, `16xlarge - 128`
- サービス:
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon MQ
  - Amazon Neptune
  - Amazon RDS

O

product/operatingSystem

- 説明: Amazon EC2 インスタンスのオペレーティングシステムを示します。
- サンプル値： `Amazon Linux`, `Ubuntu`, `Windows Server`, `Oracle Linux`, `FreeBSD`
- サービス:

- Amazon AppStream
- Amazon EC2
- Amazon GameLift Servers
- Amazon Lightsail
- Amazon WorkSpaces
- AWS CodeBuild

product/operation

- 説明: この明細項目の対象となる特定の AWS オペレーションを表します。
- サンプル値： `RunInstances` (Amazon EC2 インスタンスのオペレーションを示します)
- サービス:
  - Amazon EC2
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB
  - Amazon CloudWatch
  - Amazon Redshift

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/originationIdType

- 説明: SMS メッセージの送信時に使用される発信元 ID の種類について説明します。
- サンプル値： `Sharedroute`
- サービス:
  - Amazon Simple Notification Service

product/osType

- 説明: リソースのオペレーティング システムについて説明します。
- サンプル値： `Dev Environment`, `Linux`, `Linux ARM64`, `Windows`

- サービス:
  - Amazon CodeCatalyst

P

product/parameterType

- 説明: スタックの作成時または更新時に、パラメータを使用してテンプレートにカスタム値を入力します。 AWS CloudFormation 例えば、`InstanceTypeParameter` と指定します。このパラメータを使用して、スタックを作成または更新するときに Amazon EC2 インスタンス タイプを指定できます。

product/physicalCores

- 説明: インスタンスが提供する物理コアの数を表します。
- サンプル値： `4`, `8`
- サービス:
  - Amazon EC2

product/physicalProcessor

- 説明: Amazon EC2 インスタンス上のプロセッサを示します。
- サンプル値： `High Frequency Intel Xeon E7-8880 v3 (Haswell)`, `Intel Xeon E5-2670`, `AMD EPYC 7571`
- サービス:
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon Neptune
  - Amazon RDS
  - AWS データベース移行サービス

product/platoClassificationType

- 説明: データ注釈ワークフロー ルーティングのオブジェクトごとの段階的な価格設定。。

- サンプル値： `LabeledObject`, `3DLabeledObjectMultiFrame`, `3DLabeledObject`, `Processing:VolumeUsage`
- サービス:
  - Amazon SageMaker AI

product/pricingUnit

- 説明： AWS サービスの最小請求単位。たとえば、API 呼び出しあたり 0.01c です。
- サービス:
  - AWS Directory Service

product/primaryplaceofuse

- 説明: 顧客が主にサービスを利用する主な事業所または住宅地の住所。

product/processorArchitecture

- 説明: プロセッサアーキテクチャを表します。
- サンプル値： `32-bit`, `64-bit`
- サービス:
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon Neptune
  - Amazon RDS
  - AWS データベース移行サービス

product/processorFeatures

- 説明: インスタンスのプロセッサ機能を表します。
- サンプル値： `Intel AVX`, `Intel AVX2`, `Intel AVX512`, `Intel Turbo`
- サービス:
  - AWS データベース移行サービス
  - Amazon DocumentDB

- Amazon EC2
- Amazon Neptune
- Amazon RDS

product/ProductFamily

- 説明: 製品タイプのカテゴリ。
- サンプル値： `Alarm`, `AWS Budgets`, `Stopped Instance`, `Storage Snapshot`, `Compute`
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/ProductName

- 説明: AWS のサービスの完全な名前。この列を使用して、 AWS サービス別に AWS 使用状況をフィルタリングします。
- サンプル値： `AWS Backup`, `AWS Config`, `Amazon Registrar`, `Amazon Elastic File System`, `Amazon Elastic Compute Cloud`

product/productSchemaDescription

- 説明: 製品がどのように構築されているかを示す設計図。これには、製品を構成するさまざまな属性が含まれます。

product/provisioned

- 説明: Amazon EBS の使用状況がプロビジョニングされた Amazon EBS ストレージに関連していたかどうかを示します。

- サンプル値： `Yes`, `No`
- サービス:
  - Amazon EC2
  - Amazon MQ

product/provisioningType

- 説明: リソースがオンデマンドでデプロイされたのか、事前にプロビジョニングされたのかを説明します。
- サンプル値： `On-Demand`, `Pre-Provisioned`
- サービス:
  - Amazon CodeCatalyst

product/PurchaseOption

- 説明： AWS サービスで使用可能な購入モデルについて説明します。たとえば、 は、、`On-Demand`、 の 4 つの主要な Amazon EC2 `Reserved Instances`インスタンス購入オプションと`Spot Instances`、 の追加オプション AWS を提供します`Dedicated Hosts`。

product/purchaseterm

- 説明: Amazon EC2 では、一貫性のあるインスタンス設定へのコミットメントを指定します。これには、1 ~ 3 年間のインスタンスタイプとリージョンが含まれます。

R

product/region

- 説明： AWS サービスをホストする地理的エリア。このフィールドを使用して、特定のリージョンにわたって使用量を分析できます。
- サンプル値： `eu-west-3`, `us-west-1`, `us-east-1`, `ap-northeast-2`, `sa-east-1`
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3

- Amazon RDS
- Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/regioncode

- 説明： リージョンは、データセンターがクラスター化されている世界中の物理的な場所です。 は、論理データセンターの各グループをアベイラビリティーゾーン (AZ) と AWS 呼びます。 AWS 各リージョンは、地理的エリア内の複数の、隔離された、物理的に別々の AZ で構成されています。リージョンコード属性の名前は AWS リージョンと同じで、 AWS サービスが利用可能な場所を指定します。
- サンプル値： `us-west-2`, `us-east-1`, `ap-southeast-2`
- サービス:
  - Amazon SageMaker AI

product/replicationType

- 説明: サービスが無料で使用できることを指定します。たとえば、 AWS Server Migration Service は無料で使用でき、移行プロセス中に使用されるストレージリソースに対してのみ料金が発生します。
- サンプル値： `Free`
- サービス:
  - AWS アプリケーション移行サービス

product/resourceAssessment

- 説明: 証拠を収集、保存、管理するプロセス。これを利用して、リスクや業界標準や規制の遵守状況を評価できます。
- サンプル値： `All assessment`
- サービス:
  - AWS Audit Manager

product/resourcePriceGroup

- 説明: リソースタイプ、リソース、価格グループ (モニタリング時に請求する価格。現在 A と B の 2 つの価格クラスがあります) について説明します。したがって、例として RDS リソースを監視する場合、リソースタイプは RDS (「製品」)、リソースはインスタンス、価格グループは B になります。
- サンプル値： `RDS-DBInstance-GroupB`
- サービス:
  - Amazon DevOps Guru

product/routeType

- 説明:使用する SMS ルートのタイプについて説明します。現時点では標準のみが適用されます。
- サンプル値： `Standard`
- サービス:
  - Amazon Simple Notification Service

S

product/servicecode

- 説明： これは、顧客への特定の AWS サービスを一意の短い略語として識別します。
- サンプル値： `Amazon EC2`, `AWS KMS`
- サービス:
  - AWS 予算
  - AWS Backup
  - AWS Certificate Manager
  - AWS Cloud Map
  - AWS CloudTrail

product/servicename

- 説明： AWS サービスに関する簡単な説明。
- サービス:

- Amazon EC2 Budgets
- Amazon ECR
- Amazon ECS
- Amazon EFS
- Amazon Elastic Inference
- Amazon EKS

product/singleOrDualPass

- 説明: 動画のエンコーディングの種類を決定するために使われる用語。ワンパスエンコーディングでは、エンコーディングは最初のパス自体で行われます。2 パスエンコーディングでは、1 回目のパスでファイルが徹底的に分析され、中間ファイルが作成されます。2 番目のパスでは、エンコーダーが中間ファイルを見つけてビットを割り当てます。実際のエンコーディングは 2 回目のパスで行われます。
- サービス:
  - AWS Elemental MediaConvert

product/sizeFlex

- 説明: RI の標準化された利点をリージョンやインスタンスファミリー内の他のインスタンスサイズにも適用できるかどうかを説明します。
- サンプル値： `true`, `false`
- サービス:
  - Amazon Elastic Compute Cloud

product/sku

- 説明: 製品の一意のコード。SKU は、`ProductCode`、`UsageType`、および `Operation` を組み合わせて作成されます。サイズ柔軟な RI の場合、SKU は使用されたインスタンスを使用します。たとえば、`t2.micro`インスタンスを使用し、使用量に RI `t2.small` 割引 AWS を適用した場合、明細項目 SKU は で作成されます`t2.micro`。
- サンプル値： `FFNT87MQSCR328W6`, `VBYCEU494XUAHCA7`
- サービス:
  - Amazon EC2

- AWS Certificate Manager
- Amazon S3
- Amazon RDS
- Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/storage

- 説明: インスタンスにアタッチされているディスクストレージを表します。
- サンプル値： `60GB`, `True`, `EBS Only`, `1 x 900 NVMe SSD`, `1 x 150 NVMe SSD`
- サービス:
  - Amazon EC2
  - Amazon RDS
  - Amazon Redshift
  - OpenSearch Service
  - Amazon WorkSpaces

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/storageclass

- 説明: Amazon S3 バケットのストレージクラスを示します。
- サンプル値： `Archive`, `General Purpose`, `Infrequent Access`, `Intelligent-Tiering`, `Non-Critical Data`
- サービス:
  - AWS Elemental MediaStore
  - AWS Storage Gateway
  - Amazon Cloud Directory
  - Amazon EFS
  - Amazon MQ

- Amazon S3

product/storagemedia

- 説明: 記憶媒体とは、電子データの保存、保管、検索に使用されるデバイスや材料を含むあらゆる技術です。
- サービス:
  - AWS Database Migration Service
  - Amazon CloudWatch
  - Amazon DocumentDB
  - Amazon EC2
  - Amazon ES

product/storagetype

- 説明: コンピューターが情報を保存する方法と場所について説明します。これは、コンピューター、サーバー、またはコンピューティングデバイスの内部または外部の場合があります。
- サンプル値： `Amazon S3`, `SSD`, `SSD-backed`
- サービス:
  - AWS Backup
  - Amazon ECR

T

product/tenancy

- 説明: Amazon EC2 インスタンスで許可されているテナンシーのタイプ。
- サンプル値： `Dedicated`, `Reserved`, `Shared`, `NA`, `Host`
- サービス:
  - Amazon EC2
  - Amazon ECS

product/throughputCapacity

- 説明: ファイルシステムをホストしているファイルサーバーがファイルデータを提供できる速度について説明します。Amazon FileCache の場合、値は 1000 のみになります。
- サンプル値： `12`, `40`, `50`, `100`, `125`, `250`, `500`, `1000`
- サービス:
  - Amazon FileCache
  - Amazon FSx

product/tier

- 説明： を使用すると AWS、使用量が増えるにつれて、ボリュームベースの割引と割引を受けることができます。Amazon S3 のようなサービスの場合、料金は段階的です。つまり、使用量が多いほど、GB あたりの支払いが少なくなります。 AWS は、ビジネスニーズを支援するサービスを取得するためのオプションを提供します。
- サービス:
  - AWS Elemental MediaConvert

product/toLocation

- 説明: 使用先の場所を示します。
- サンプル値： `External`, `US East (N. Virginia)`
- サービス:
  - Amazon CloudFront
  - AWS データ転送

product/toLocationType

- 説明: サービス使用先の場所のタイプを示します。
- サンプル値： `AWS Region`, `AWS Edge Location`
- サービス:
  - AWS Direct Connect
  - AWS Elemental MediaConnect
  - AWS Shield

- Amazon CloudFront
- Amazon Lightsail

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/toRegionCode

- 説明： AWS サービスのソースリージョンコードについて説明します。詳細については、「product/regioncode」を参照してください。
- サンプル値： `eu-west-1`
- サービス:
  - Amazon RDS
  - Amazon EC2
  - Amazon VPC
  - AWS Direct Connect

product/transcodingResult

- 説明: エンコードされたビデオソースを中間の非圧縮形式にデコードし、ターゲット形式に再エンコードする場合の出力。
- サービス:
  - AWS Elemental MediaConvert

product/trialProduct

- 説明: AWS CloudHSM が空き時間を許容するかどうかを説明します。
- サービス:
  - AWS CloudHSM

U

product/upfrontCommitment

- 説明： 使用コミットメントが必要かどうかを記述します AWS CloudHSM。HSM が AWS CloudHSM クラスターにプロビジョニングされる 1 時間 (またはその時間の一部) ごとに、時間単位の料金が請求されます。HSM のないクラスターには課金されず、暗号化されたバックアップの自動保存についても請求されません。詳細については、AWS CloudHSM の料金を参照してください。

  HSM へのネットワークデータ転送と HSM からのネットワークデータ転送には別途料金がかかります。詳細については、Amazon EC2 の料金表を参照してください。
- サービス:
  - AWS CloudHSM

product/usagetype

- 説明: 明細項目の使用状況の詳細を表します。
- サンプル値： `EU-BoxUsage:c5d.9xlarge`, `EU-BoxUsage:m4.16xlarge`, `SAE1-InstanceUsage:db.t2.medium`, `USW2-AW-SW-19`, `SAE1-BoxUsage:c4.large`,
- サービス:
  - Amazon EC2
  - AWS Certificate Manager
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

V

product/vcpu

- 説明: 単一の CPU コア上で同時に実行されているスレッドの数を示します。Amazon EC2 インスタンスは、単一の Intel Xeon CPU コアで同時に複数のスレッドを実行できるマルチスレッドをサポートしています。各スレッドは、インスタンスの仮想 CPU (vCPU) として表されます。
- サンプル値： 8, 16, 36, 72, 128
- サービス:
  - Amazon EC2
  - Amazon RDS
  - Amazon Redshift
  - OpenSearch Service
  - Amazon ElastiCache

  詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

product/videoCodec

- 説明: デジタルビデオを圧縮および解凍するソフトウェアまたはハードウェア。ビデオ圧縮では、コーデックはエンコーダーとデコーダーを組み合わせたものです。圧縮のみを行うデバイスは一般にエンコーダーと呼ばれ、解凍のみを行うデバイスはデコーダーと呼ばれます。
- サービス:
  - AWS Elemental MediaConvert;

product/videoFrameRate

- 説明: ビデオフレームレート (1 秒あたりのフレーム数 (FPS) で表示) は、ビデオカメラ、コンピュータグラフィックス、およびモーションキャプチャシステムによって連続した画像 (フレーム) がキャプチャまたは表示される周波数レートです。
- サービス:
  - AWS Elemental MediaConvert;

product/videoQualitySetting

- 説明: エンコードに使用される品質設定について説明します。これは圧縮効率、ひいては特定のビットレートでのビデオ品質に影響します。
- サンプル値： `Multi-pass`, `Multi-pass HQ`, NA, `Single-pass`, `Single-pass HQ`,
- サービス:
  - AWS Elemental MediaConvert

product/volumeType

- 説明: Amazon EBS のボリュームタイプを示します。
- サンプル値： `Standard`, `General Purpose`, `General Purpose-Aurora`, `Amazon Glacier`, `Amazon SimpleDB – Standard`,
- サービス:
  - Amazon EC2
  - Amazon S3
  - Amazon RDS
  - Amazon DynamoDB
  - Amazon S3 Glacier

    詳細。完全なサービスリストについては、Column_Attribute_Service.zip をダウンロードしてください。

W

product/workforceType

- 説明: 雇用されている労働者と失業者の労働力プールの細分化。例えば、`Full Time Employees` (FTE)、または `Temporary`。
- サービス:
  - Amazon SageMaker AI

リソースタグの詳細

AWS コストと使用状況レポートのresource列を使用して、明細項目の対象となる特定のリソースに関する情報を検索できます。これらの列には、ユーザー定義のコスト配分タグが含まれます。次に例を示します。

resourceTags/user:creator

user:Creator タグを使用して、リソースを作成したユーザーを追跡します。

resourceTags/user:name

user:Name タグを使用して、特定のユーザーに関連付けられているリソースを追跡します。

resourceTags/user:owner

user:Owner タグを使用して、リソースを所有しているユーザーを追跡します。

resourceTags/user:purpose

user:Purpose タグを使用して、リソースが作成された理由を追跡します。

Savings Plans の詳細

AWS コストと使用状況レポートのSavingsPlan列には、Savings Plans の詳細が表示されます。Savings Plans の詳細については、Savings Plans ユーザーガイド の Savings Plans とは？ を参照してください。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

A

savingsPlan/AmortizedUpfrontCommitmentForBillingPeriod

- 説明: 請求期間に Savings Plan サブスクリプションに必要となる前払い金額。[全額前払い Savings Plan] と [一部前払い Savings Plan] の最初の前払い金は、当月に償却されます。[前払いなし Savings Plan] の場合、値は 0 です。
- 該当する明細項目: SavingsPlanRecurringFee
- サービス:

- Amazon EC2
- Fargate
- AWS Lambda
- Amazon SageMaker AI

E

savingsPlan/EndTime

- 説明: Savings Plan 契約の有効期限日。
- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

I

savingsPlan/InstanceTypeFamily

- 説明: 指定した使用量に関連するインスタンスファミリー。
- 該当する明細項目: SavingsPlanCoveredUsage
- サンプル値： `m4, g2`
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

N

savingsPlan/NetAmortizedUpfrontCommitmentForBillingPeriod

請求期間中の Savings Plan サブスクリプションの前払い料金のコスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

savingsPlan/NetRecurringCommitmentForBillingPeriod

Savings Plan 手数料の混合されていない正味費用。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

savingsPlan/NetSavingsPlanEffectiveCost

Savings Plans の実効費用。使用量を手数料で割ったものです。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。

O

savingsPlan/OfferingType

- 説明: 購入した Savings Plan のタイプを示します。
- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サンプル値： `ComputeSavingsPlans`, `EC2InstanceSavingsPlans`, `SageMakerSavingsPlans`
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

P

savingsPlan/PaymentOption

- 説明: Savings Plan に使用できる支払いオプション。

- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サンプル値： `Partial Upfront`, `All Upfront`, `No Upfront`
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

savingsPlan/PurchaseTerm

- 説明: Savings Plan の期間を示します。
- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サンプル値： `1yr`, `3yr`
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

R

savingsPlan/RecurringCommitmentForBillingPeriod

- 説明: Savings Plan サブスクリプションの毎月の定額料金。例えば、[一部前払い Savings Plan] や [前払いなし Savings Plan] の毎月の定額料金が該当します。
- 該当する明細項目: SavingsPlanRecurringFee
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

savingsPlan/Region

- 説明： AWS サービスをホストする AWS リージョン (地理的エリア) 。このフィールドを使用して、特定の AWS リージョンの支出を分析できます。
- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サンプル値： `US East (N. Virginia)`, `US West (N. California)`, `US East (Ohio)`, `Asia Pacific (Mumbai)`, `Europe (Ireland)`
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

S

savingsPlan/SavingsPlanArn

- 説明: Savings Plan の一意の ID。
- 該当する明細項目: SavingsPlanUpfrontFee
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

savingsPlan/SavingsPlanEffectiveCost

- 説明: 各使用量の行に割り当てられている Savings Plan 月額料金 (前払いおよび定額) の比率。
- 該当する明細項目: SavingsPlanCoveredUsage
- サービス:
  - Amazon EC2
  - Fargate

- AWS Lambda
- Amazon SageMaker AI

savingsPlan/SavingsPlanRate

- 説明: 使用量に対する Savings Plan の料金。
- 該当する明細項目: SavingsPlanCoveredUsage
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

savingsPlan/StartTime

- 説明: Savings Plan 契約の開始日。
- 該当する明細項目: SavingsPlanCoveredUsage、SavingsPlanNegation、SavingsPlanUpfrontFee、SavingsPlanRecurringFee
- サービス:
  - Amazon EC2
  - Fargate
  - AWS Lambda
  - Amazon SageMaker AI

T

savingsPlan/TotalCommitmentToDate

- 説明: 今日の該当時間までに償却された前払いコミットメントと定額コミットメントの合計。
- 該当する明細項目: SavingsPlanRecurringFee
- サービス:
  - Amazon EC2
  - Fargate

- AWS Lambda
- Amazon SageMaker AI

U

savingsPlan/UsedCommitment

- 説明: 使用している Savings Plan コミットメントの USD 総額。(SavingsPlanRate と使用量を乗じた額)
- 該当する明細項目: SavingsPlanRecurringFee
- サービス:
  - Amazon EC2
  - AWS Lambda
  - Fargate
  - Amazon SageMaker AI

Cost Categoriesの 詳細

AWS コストと使用状況レポートのCost Category列を使用して、Cost Categories機能によって分類された特定の明細項目に関する情報を検索できます。 Cost Category列は、コストカテゴリと分類ルールを作成するときに自動的に入力されます。これらの列には、列ヘッダーとしてユーザー定義の Cost Category 名と、対応する Cost Category 値が含まれます。次に例を示します。

costCategory/Project

Project Cost Category を使用して、プロジェクトに関連付けられたコストと使用量の情報を分類します。「alpha」や「beta」などのプロジェクト名は、costCategory/Projectの下に値として表示されます。

costCategory/Team

Team Cost Category を使用して、チームに関連付けられたコストと使用量の情報を分類します。「Marketing」などのチーム名は、costCategory/Team の下に値として表示されます。

costCategory/Environment

Environment Cost Category を使用して、環境に関連付けられたコストと使用量の情報を分類します。「Development」などの環境名は、costCategory/Environment の下に値として表示されます。

ディスカウントの詳細

discountヘッダーの下の列は、レポートの請求期間中にアカウントにdiscount AWS が適用されている場合にのみコストと使用状況レポートに含まれます。これには以下の列が含まれます。

discount/BundledDiscount

ラインアイテムに適用されるバンドルdiscount。バンドルdiscount は、サービスまたは機能を別のサービスまたは機能の使用量に基づいて無料またはdiscount 料金で提供する使用量ベースのディスカウントです。バンドルディスカウントの例には以下が含まれます。

- を使用する場合 AWS Shield Advanced、別途料金 AWS WAF を支払う必要はありません。 AWS WAF 使用量は にバンドルされています AWS Shield Advanced。詳細については AWS Shield Advanced、Amazon CloudFront の料金」を参照してください。
- を使用して NAT ゲートウェイを作成する場合 AWS Network Firewall、標準の NAT ゲートウェイ処理と 1 時間あたりの使用料金は、GB あたりのファイアウォール処理と使用時間に基づいて one-to-one で免除されます。詳細については、「AWS Network Firewall 料金表」を参照してください。
- Amazon インタラクティブビデオサービス (IVS) チャットでは、ビデオ入力が 1 時間送信されるごとに、2,700 通のメッセージが送信され、27万件のメッセージが配信されます。追加料金は発生しません。詳細については、「Amazon Service Video Service 料金」を参照してください。

discount/TotalDiscount

対応する品目のすべてのdiscount 列の合計。

分割明細項目の詳細

AWS コストと使用状況レポートの splitLineItemヘッダーの列は、分割コスト配分データ機能をオプトインした場合にコストと使用状況レポートに表示されるフィールドです。詳細については、「分割コスト配分データについて」を参照してください。この機能は、Amazon ECS (Fargate を含む) AWS Batch、および Amazon EKS のみに制限されています。

A| B | C | D | E | F | G | H | I | J | K | L | M | | O N | | Q P | | T R | S | | V U | W | X | Y | Z

A

splitLineItem/ActualUsage

- 説明: Amazon ECS タスクまたは Kubernetes ポッドで指定した期間に発生した vCPU またはメモリの使用量 (lineItem/UsageType に基づく)。
- 該当する明細項目: Usage
- サンプル値： 0.1, 0.5, 1.3
- サービス: Amazon ECS、Fargate、Amazon EKS

> **Note**
>
> Fargate のコストは vCPU とメモリの予約に基づいて計算され、lineItem/UsageAmount 列に反映されます。スプリットコスト配分データは、低レイテンシーのテレメトリデータを観察し、それを時間、日、月に集計することにより、実際の Fargate vCPU とメモリの使用量を計算します。そのため、 splitLineItem/ActualUsage は lineItem/UsageAmount とは異なる場合があります。

N

splitLineItem/NetSplitCost

- 説明: すべてのディスカウントが適用された後の Amazon ECS タスクまたは Kubernetes ポッドの実効コスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。
- 該当する明細項目: Usage
- サンプル値： 1.35, 1.75
- サービス: Amazon ECS、Fargate、Amazon EKS

splitLineItem/NetUnusedCost

- 説明: すべてのディスカウントが適用された後の Amazon ECS タスクまたは Kubernetes ポッドの実効未使用コスト。この列は、該当する請求期間中にアカウントにディスカウントがある場合にのみレポートに含まれます。
- 該当する明細項目: Usage

- サンプル値： 1.35, 1.75
- サービス: Amazon ECS、Fargate、Amazon EKS

> **Note**
> 未使用のコストは、splitLineItem/SplitUsage に基づいて Amazon ECS タスクまたは Kubernetes ポッドに比例して適用されます。

P

splitLineItem/ParentResourceId

- 説明: Amazon ECS タスクまたは Amazon EKS ポッドに関連付けられている親 EC2 インスタンスのリソース ID (lineItem/ResourceId 列で参照されています)。親リソース ID は、指定された期間の ECS タスクまたは Kubernetes ポッドのワークロードが親 EC2 インスタンスで実行されたことを意味します。これは、起動タイプが EC2 の Amazon ECS タスクまたは Kubernetes ポッドにのみ適用されます。
- 該当する明細項目: Usage
- サービス: Amazon ECS、Amazon EKS

> **Note**
> splitLineItem/ParentResourceId は、リソース IDsが AWS コストと使用状況レポートに含まれている場合にのみ使用できます。

splitLineItem/PublicOnDemandSplitCost

- 説明: パブリックオンデマンドインスタンス料金 (pricing/publicOnDemandRate 列参照) に基づく、その期間に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリのコスト (lineItem/UsageType に基づく)。
- 該当する明細項目: Usage
- サンプル値： 1.35, 1.75
- サービス: Amazon ECS、Fargate、Amazon EKS

splitLineItem/PublicOnDemandUnusedCost

- 説明: パブリックオンデマンドインスタンス料金に基づいて、Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリの未使用コスト (lineItem/UsageType に基づく)。未使用コストとは、EC2 インスタンス (splitLineItem/ParentResourceId 列で参照) のリソース (CPU またはメモリ) で、指定された期間に使用されなかったコストです。
- 該当する明細項目: Usage
- サンプル値： 1.35, 1.75
- サービス: Amazon ECS、Fargate、Amazon EKS

R

splitLineItem/ReservedUsage

- 説明: Amazon ECS タスクまたは Kubernetes ポッドで指定した期間に設定した vCPU またはメモリの使用量 (lineItem/UsageType に基づく)。
- 該当する明細項目: Usage
- サンプル値： 1, 2, 4
- サービス: Amazon ECS、Fargate、Amazon EKS

S

splitLineItem/SplitCost

- 説明: 期間中に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリのコスト (lineItem/UsageType に基づく)。これには、EC2 インスタンス (splitLineItem/parentResourceId 列で参照) に予約またはSavings Plans の前払いまたは一部前払い料金がある場合の償却費用が含まれます。
- 該当する明細項目: Usage
- サンプル値： 1.35, 1.75
- サービス: Amazon ECS、Fargate、Amazon EKS

splitLineItem/SplitUsage

- 説明: 指定した期間に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリ (lineItem/UsageType に基づく) の使用量。これは splitLineItem/ReservedUsage または splitLineItem/ActualUsage の最大使用量として定義されます。
- 該当する明細項目: Usage
- サンプル値： 1, 1.3
- サービス: Amazon ECS、Fargate、Amazon EKS

splitLineItem/SplitUsageRatio

- 説明: Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリと、EC2 インスタンス (splitLineItem/ParentResourceId 列で参照) で使用可能な CPU またはメモリ全体の比率 (lineItem/UsageType に基づく)。
- 該当する明細項目: Usage
- サンプル値： 0.25, 0.75
- サービス: Amazon ECS、Fargate、Amazon EKS

> **Note**
>
> splitLineItem/SplitUsageRatio は、時間単位のデータを優先する AWS のコストと使用状況レポートでのみ使用できます。

U

splitLineItem/UnusedCost

- 説明: 期間中に Amazon ECS タスクまたは Kubernetes ポッドに割り当てられた vCPU またはメモリの未使用コスト (lineItem/UsageType に基づく)。未使用コストとは、EC2 インスタンス (splitLineItem/ParentResourceId 列で参照) のリソース (CPU またはメモリ) で、指定された期間に使用されなかったコストです。これには、EC2 インスタンス (splitLineItem/parentResourceId) に予約または Savings Plans の前払いまたは一部前払い料金がある場合の償却費用が含まれます。
- 該当する明細項目: Usage
- サンプル値： 1.35, 1.75

- サービス: Amazon ECS、Fargate、Amazon EKS

> **Note**
> 未使用のコストは、splitLineItem/SplitUsage に基づいて Amazon ECS タスクまたは Kubernetes ポッドに比例して適用されます。

ユースケース

AWS コストと使用状況レポート (AWS CUR) は、レポート管理のニーズに合わせて使用できます。このセクションでは、Savings Plans やリザーブドインスタンス (RI) の使用率、料金、割り当ての追跡などのユースケースについて詳しく説明します。

トピック

- Savings Plans について
- 予約を理解する
- データ転送料金について
- 分割コスト配分データを理解する

Savings Plans について

コストと使用状況レポート (AWS CUR) を使用して、Savings Plans の使用率、料金、配分を追跡できます。

Savings Plans 明細項目

Savings Plans は、Amazon EC2、、Amazon SageMaker AI を低価格で提供する柔軟な料金モデルを提供します。これと引き換えに AWS Fargate AWS Lambda、1 年または 3 年の期間にわたって一貫した使用量 ($/時間で測定) を約束します。

CUR の次の明細項目は、Savings Plans AWS での支出の追跡と管理に役立ちます。

> **Note**
> 次の表では、わかりやすくするために CUR AWS の列と行が変換されています。最初の列の値はレポートのヘッダーを表します。これらの例には、いくつかのキー CUR AWS 列のみが

含まれます。他の CUR AWS 列の詳細については、「」を参照してください[データディクショナリ](#)。

前払い料金

[SavingsPlanUpfrontFee] 明細項目は、`All Upfront` または `Partial Upfront` Savings Plan を購入すると、請求書に追加されます。次の表は、この 1 回限りの料金が一部の CUR AWS 列にどのように表示されるかを示しています。

lineItem/LineItemType	SavingsPlanUpfrontFee	SavingsPlanUpfrontFee
lineItem/ProductCode	ComputeSavingsPlans	ComputeSavingsPlans
lineItem/UsageStartDate	2019-10-10T00:03:54Z	2019-10-10T00:12:15Z
lineItem/LineItemDescription	1 年間の全額前払いコンピューティング Savings Plan ID: 70352035 に対する 1 回限りの料金 43.80 USD	3 年間の一部前払いリージョン M5 インスタンスタイプ EC2 Savings Plan ID: 12355516 に対する 1 回限りの料金 43.80 USD
lineItem/UnblendedCost	43.8	43.8
savingsPlan/SavingsPlanARN	arn:aws:savingsplans::5555555555555:savingsplan/bc1d08fd	arn:aws:savingsplans::5555555555555:savingsplan/67b0ef20

Savings Plans の毎月の定額料金

[SavingsPlanRecurringFee] 明細項目は、`No Upfront` または `Partial Upfront` Savings Plans に対応する毎時間の定額料金を示します。最初は、SavingsPlanRecurringFeeが購入日とその後は 1 時間ごとに請求書に追加されます。

[SavingsPlanRecurringFee] は、時間 (毎時間のコストと使用量に適用) または日 (毎日のコストと使用量に適用) に割り当てられ、購入した時刻に請求書に追加されます。以降は、請求期間の時間/日ごとに追加されます。

`All Upfront`Savings Plan の場合、明細項目には、請求期間中に未使用の Savings Plan の部分が表示されます。

次の表は、定期的な時間単位の料金が一部の CUR AWS 列にどのように表示されるかを示しています。

lineItem/LineItemType	SavingsPlanRecurringFee	SavingsPlanRecurringFee
lineItem/UsageStartDate	2019-20-10T00:00:00Z	2019-20-10T00:00:00Z
lineItem/ProductCode	Compute Savings Plans	Compute Savings Plans
lineItem/UsageType	ComputeSP:1yrPartialUpfront	USE2-EC2SP:t3.1yrPartialUpfront
lineItem/UnblendedCost	0.01	0.01
lineItem/LineItemDescription	1 year Partial Upfront Compute Savings Plan	1 year Partial Upfront t3 EC2 Instance Savings Plan in us-east-2
savingsPlan/SavingsPlanARN	arn: aws: savingsplans:: 555555555555555555: savingsplan/ bc1d08fd	arn: aws: savingsplans:: 555555555555555: savingsplan/ bc1d08fd
savingsPlan/PaymentOption	一部前払い	一部前払い
savingsPlan/OfferingType	ComputeSavingsPlans	EC2InstanceSavingsPlans
savingsPlan/PurchaseTerm	1yr	1yr

> **Note**
>
> SSavingsPlanRecurringFee は、定期的なRI手数料とは異なる方法で計算されます。定期的な RI 料金は月額料金ですが、SSavingsPlanRecurringFee は1時間ごとの料金です。定期的な RI 料金については、定額月額 RI 料金を参照してください。

Savings Plans 割引特典

[SavingsPlanCoveredUsage] 明細項目は、Savings Plans の特典が適用されたインスタンス使用量を示します。[SavingsPlanCoveredUsage] 明細項目は、Savings Plan 特典が適用されなかった場合のオンデマンド料金の非ブレンドコストを示します。この非ブレンドコストは、対応する [SavingsPlanNegation] 明細項目で相殺されます。

SavingsPlanCoveredUsage の各明細項目では、savingsPlan/SavingsPlanRateおよびsavingsPlan/SavingsPlanEffectiveCostフィールドを使用して、Savings Plans の時間単位のコミットメントに対してその使用量がどのように請求されたかを確認できます。

SavingsPlanCoveredUsage 明細項目ごとに、対応する SavingsPlanNegation を確認できます。[SavingsPlanNegation] 明細項目は、[SavingsPlanCoveredUsage] の非ブレンドコストを相殺し、SavingsPlanARN、オペレーション、使用タイプ、およびアベイラビリティーゾーン別に時間レベルでグループ化されます。このため、1 つの SavingsPlanNegation 明細項目が複数の SavingsPlanCoveredUsage 明細項目に対応する場合があります。

次の表は、一部の CUR AWS 列で対象使用量と否定行項目がどのように表示されるかを示しています。

lineItem/LineItemType	SavingsPlanCoveredUsage	SavingsPlanCoveredUsage	SavingsPlanNegation
lineItem/UsageStartDate	2019-10-10T00:00:00Z	2019-10-10T00:00:00Z	2019-10-10T00:00:00Z
lineItem/UsageEndDate	2019-10-10T01:00:00Z	2019-10-10T01:00:00Z	2019-10-10T01:00:00Z
lineItem/ProductCode	AmazonEC2	AmazonEC2	AmazonEC2
lineItem/UsageType	BoxUsage:t3.nano	BoxUsage:t3.nano	BoxUsage:t3.nano
lineItem/UsageAmount	1	0.5	-1.5

lineItem/ UnblendedCost	0.0052	0.0026	-0.0078
lineItem/ LineItemDescription	$0.0052 per On Demand Linux t3.nano Instance Hour	$0.0052 per On Demand Linux t3.nano Instance Hour	SavingsPlanNegation used by AccountId : 5555555555555 and UsageSku : K7ERD2Q28HHU97DT
SavingsPlan/ SavingPlanARN	arn:aws:savingsplans:: 5555555555555: savingsplan/bc1d08fd	arn:aws:savingsplans:: 5555555555555: savingsplan/bc1d08fd	arn:aws:savingsplans:: 5555555555555: savingsplan/bc1d08fd
savingsPlan/ SavingsPlanRate	0.0026	0.0026	
savingsPlan/ SavingsPlanEffectiveCost	0.0026	0.0013	

使用量が、Savings Plans コミットメントでカバーできる範囲を超えた場合、カバーされない使用量は依然として [Usage] (使用量) 明細項目として表示され、カバーされた使用量は [SavingsPlanCoveredUsage] として、対応する [SavingsPlanNegation] 明細項目と共に表示されます。

予約を理解する

AWS コストと使用状況レポート (AWS CUR) を使用して、リザーブドインスタンス (RI) の使用率、料金、割り当てを追跡できます。このセクションでは、予約について詳しく説明します。

トピック

- 予約の明細項目を理解する
- 償却された予約データを理解する
- Amazon EC2 のサイズフレキシブル予約の監視
- オンデマンドキャパシティー予約のモニタリング

予約の明細項目を理解する

RI では、オンデマンドインスタンスの料金と比べて大幅な割引を受けられます。RI は物理インスタンスではありません。アカウントでのオンデマンドインスタンスの使用に適用される請求割引です。これらのオンデマンドインスタンスは、請求の割引のメリットを得るために特定の属性に一致する必要があります。

トピック

- 前払い料金
- 調整手数料
- 定額月額 RI 料金
- RI 割引特典
- Reserved Instance type
- リザーブドインスタンスのメリットはインスタンスの使用に適用されます

> **Note**
>
> 次の表では、わかりやすくするために CUR AWS の列と行が変換されています。最初の列の値はレポートのヘッダーを表します。これらの例には、いくつかのキー CUR AWS 列のみが含まれます。他の CUR AWS 列の詳細については、「」を参照してくださいデータディクショナリ。

前払い料金

[料金] 明細項目は、RI の `All Upfront` または `Partial Upfront` を購入した場合に請求書に追加されます。

次の表は、この 1 回限りの料金が一部の CUR AWS 列にどのように表示されるかを示しています。

lineItem/LineItemType	料金
lineItem/ProductCode	AmazonEC2
lineItem/UsageStartDate	2016-01-01T00:00:00Z

lineItem/LineItemDescription	Sign up charge for subscription: 363836886, planId: 1026576
lineItem/UnblendedCost	68
Reservation/ReservationARN	arn:aws:ec2:us-east-1:123456789012:reserved-instances/f8c204c1-dd48-43f1-adb8-f88aa61e0dea

調整手数料

コンバーティブルリザーブドインスタンスを交換する場合、元のリザーブドインスタンスと新しいリザーブドインスタンスの交換に関連する費用 (調整手数料) も、料金明細項目として請求書に追加されます。調整手数料については、reservation/ReservationARN 列は reserved-instance-exchange/riexを含んでいます。

次の表は、コンバーティブルリザーブドインスタンスを交換する際の調整手数料を示しています。

lineItem/LineItemType	lineItem/ProductCode	lineItem/UsageStartDate	lineItem/LineItemDescription	lineItem/UnblendedCost	Reservation/ReservationARN
料金	AmazonEC2	2016-01-01T00:00:00Z			arn:aws:ec2:eu-west-1:012345678901:reserved-instance-exchange/riex-examplef-5d71-4215-886f-17a3f64ea972

定額月額 RI 料金

[RI 料金] 明細項目には、その月に適用された、関連付けられた RI の定額月額料金が表示されます。[料金] は、まず購入日の請求書に追加され、そしてそれ以降の各請求期間の最初の日に追加されます。

[RI 料金] は、割引された時間料金を月ごとの時間数で乗算して計算されます。

次の表は、定期的な月額料金がどのようにレポートに表示されるかを示しています。

lineItem/LineItemType	RI 料金
lineItem/ProductCode	AmazonEC2
lineItem/UsageStartDate	2016-01-01T00:00:00Z
lineItem/UsageType	HeavyUsage: m4.large
lineItem/LineItemDescription	USD 0.0309 hourly fee per Linux/UNIX (Amazon VPC), m4.large instance
lineItem/NormalizationFactor	4
lineItem/UnblendedCost	23
Reservation/AvailabilityZone	
Reservation/ReservationARN	arn:aws:ec2:us-east-1:123456789012:reserved-instances/f8c204c1-dd48-43f1-adb8-f88aa61e0dea
Reservation/TotalReservedunits	744
Reservation/TotalReservedNormalizedUnits	2976

アベイラビリティーゾーンまたは AWS リージョン リージョンの範囲を持つ RIs では、月額料金の定期的な記録が異なります。アベイラビリティーゾーンを範囲とする RI の場合、対応するアベイラビリティーゾーンは reservation/AvailabilityZone列に表示されます。リージョン範囲がある RI の場合、[reservation/AvailabilityZone] の列は空になります。リージョン範囲がある RI は、インスタンス

サイズを示す lineitem/NormalizationFactor および reservation/TotalReservedNormalizedUnits 列の値を持っています。

> **Note**
> 定期的なRI手数料は、SavingsPlanの定期手数料とは異なる方法で計算されます。定期的な RI 料金は月額料金ですが、SSavingsPlanRecurringFee は1時間ごとの料金です。SavingsPlanの定期手数料については、を参照してください。Savings Plans について

RI 割引特典

割引使用量の明細項目は、一致する RI 割引特典を受け取ったインスタンスの使用状況を記述し、RIs のいずれかに一致する使用量がある場合に請求書に追加されます。 は、一致する使用量に基づいて RI 割引特典 AWS を計算します。たとえば、インスタンス予約に一致するインスタンスの使用などです。対応する使用料がある場合、使用料の明細項目に関連するコストは常に 0 となります。これは、RI に関連する料金がすでにそのほかの 2 つの明細項目で課金されているためです (前払い料金と定期的な月額料金)。

次の表は、RI の割引特典が適用された使用料の例を示しています。

lineItem/LineItemType	DiscountedUsage
lineItem/ProductCode	AmazonEC2
lineItem/UsageStartDate	2016-01-01T00:00:00Z
lineItem/UsageType	BoxUsage:m4.large
lineItem/LineItemDescription	Linux/UNIX (Amazon VPC), m4.large Reserved Instance applied
lineItem/ResourceId	i-1bd250bc
lineItem/AvailabilityZone	us-east-1b
lineItem/NormalizationFactor	4
lineItem/NormalizedUsageAmount	4

lineItem/UnblendedRate	0
lineItem/UnblendedCost	0
Reservation/ReservationARN	arn:aws:ec2:us-east-1:123456789012:reserved-instances/f8c204c1-dd48-43f1-adb8-f88aa61e0dea

Amazon EC2 の [DiscountedUsage] 列にある [UsageAmount] の値が実際の使用時間数です。[NormalizedUsageAmount] の値は、[UsageAmount] の値を [NormalizationFactor] の値で乗算した値です。[NormalizationFactor] の値は、インスタンスのサイズによって決まります。RI の特典である割引が対応する使用料の明細項目に適用された場合、reservation/ReservationARN列に表示される最初の前払い料金と定期的な月額料金の Amazon リソースネーム (ARN) の値は、割引使用料の明細項目の ARN の値と一致します。

正規化因子へのインスタンスサイズのマッピングの詳細については、「Amazon EC2 ユーザーガイド」の「インスタンスサイズの変更のサポート」を参照してください。

Reserved Instance type

レポートの明細項目がスタンダードリザーブドインスタンスとコンバーティブルリザーブドインスタンスのどちらに関連付けられているかを判断するには、そのlineItem/LineItemType 列を 料金 または RI 料金 でフィルタリングします。次に、product/OfferingClassリザーブドインスタンスタイプを示す列を確認します。

レポート項目が地域別リザーブドインスタンスと地域別リザーブドインスタンスのどちらに関連しているかを判断するには、reservation/AvailabilityZone列を確認してください。ゾーン型リザーブドインスタンスの場合、この列には対応するアベイラビリティーゾーンが表示されます。リージョンリザーブドインスタンスの場合、この列は空です。

リザーブドインスタンスのメリットはインスタンスの使用に適用されます

どのインスタンス使用量項目がどのリザーブドインスタンスから恩恵を受けたかを把握するには、レポートを次の 1 つまたは複数の列でフィルタリングできます。

- reservation/reservationARN: この列を予約 ARN でフィルタリングして、各項目にどのリザーブドインスタンスリースが関連付けられているかを識別します。
- lineitem/ResourceId: この列を確認して、リザーブドインスタンスの対象となるリソースの ID を確認してください。

- lineitem/LineItemType: この列を「手数料」、「RI 手数料」、または「DiscountedUsage」で絞り込むと、関連する手数料や特典を確認できます。
- lineitem/UsageType: この列を HeavyUsage でフィルタリングすると、RI 料金明細項目を識別できます。または、この列を BoxUsage でフィルタリングして、DiscountedUsage 明細項目を識別します。
- lineitem/UsageAmount: RI 料金の明細項目の場合、この列にはリザーブドインスタンスが適用された月の合計時間数が表示されます。DiscountedUsage 明細項目の場合、この列には、レポートの設定方法に応じて、日単位または月単位でリザーブドインスタンスが特定のインスタンスに適用された合計時間数が表示されます。

サイズフレキシブルなリザーブドインスタンスでインスタンスの使用量に適用される標準化された単位数を把握するには、レポートの lineitem/NormalizedUsageAmount 列を確認してください。この列の値は、以下の列の積と等しくなります。

- lineitem/UsageAmount: この列には、従量制インスタンスの使用量が時間単位で表示されます。
- lineItem/NormalizationFactor: DiscountedUsage と RI 料金の明細項目の場合、この列にはインスタンスに関連する正規化係数が表示されます。正規化因子の詳細については、「Amazon EC2 ユーザーガイド」の「正規化因子によって決定されるインスタンスサイズの柔軟性」を参照してください。

複数のアカウント AWS Organizations を持つ の場合、リザーブドインスタンスを購入または恩恵を受けたアカウントを確認するには、次の列を確認してください。

- reservation/reservationARN: 予約 ARN を確認して、どのアカウントがリザーブドインスタンスを購入したかを確認します。ARN にはアカウント ID が含まれます。
- lineitem/UsageAccountId: DiscountedUsage 明細項目の場合、この列には購入したリザーブドインスタンスから特典を受けたアカウント ID が示されます。

> **Note**
>
> リザーブドインスタンスは請求サブスクリプションであり、Amazon EC2 インスタンスのようなリソースではありません。そのため、タグが付けられたリザーブドインスタンスは、タグ付けされたリソースのように明細項目を追加しません。DiscountedUsage の明細項目の場合、タグはリザーブドインスタンスではなくタグ付けされたリソースに設定されます。

特定のリザーブドインスタンスリースに関連する費用を特定するには、料金 または RI 料金の明細項目をリース ID であるリザーブドインスタンス ARN でフィルタリングできます。リザーブドインスタンスのコストデータを整理するには、 AWS Cost Categories の使用を検討してください。詳細については、「 AWS Billing ユーザーガイド」の AWS Cost Categoriesを使用したコストの管理」を参照してください。

償却された予約データを理解する

償却とは、1 回限りの予約をそのコストの影響を受ける請求期間で分散することです。償却により、キャッシュベースの会計とは異なり、加算ベースの会計としてコストを表示できます。たとえば、1 年間の全前払いの RI で 365 USD を支払い、この RI を使用した合致するインスタンスがある場合、このインスタンスは償却されて 1 日に 1 USD のコストとなります。

請求とコスト管理で償却コストの計算に使用するデータは、以下のコストと使用状況レポート列に表示されます。

トピック

- リザーブドインスタンスのインベントリ
- 請求期間の償却データ
- リザーブドインスタンスの実質的なコスト

> **Note**
>
> すべてのリザーブドインスタンスの明細項目にすべての reservation/ 列が入力されるわけではありません。レポート内のreservation/ 列は、明細項目の種類に基づいて入力されます。たとえば、RI 料金 の明細項目が reservation/UnusedAmortizedUpfrontFeeForBillingPeriod 列に入力されます。その間、「DiscountedUsage」明細項目が reservation/effectivecost 列に表示されます。

リザーブドインスタンスのインベントリ

次の列を使用して、RI のインベントリを追跡できます。これらの列の値は RI サブスクリプション明細項目 (`RI Fee` 明細項目) のみを表示し、RI を使用した実際のインスタンス分は表示しません。

列の説明とサンプル値の詳細については、「the section called “予約の詳細”」を参照してください。

- reservation/UpfrontValue
- reservation/startTime
- reservation/endTime
- reservation/modificationStatus

請求期間の償却データ

次の列を使用すると、請求期間の RI の償却コストを理解できます。これらの列の値は RI サブスクリプション明細項目 (`RI Fee` 明細項目) のみを表示し、RI を使用した実際のインスタンス分は表示しません。

列の説明とサンプル値の詳細については、「the section called “予約の詳細”」を参照してください。

- reservation/amortizedUpfrontFeeForBillingPeriod
- reservation/unusedQuantity
- reservation/unusedNormalizedUnitQuantity
- reservation/unusedRecurringFee
- reservation/unusedAmortizedUpfrontFeeForBillingPeriod

リザーブドインスタンスの実質的なコスト

次の列を使用すると、インスタンスレベルで実質的なコストを理解できます。この列の値は、インスタンス使用状況明細項目 (`Discounted Usage boxUsage` 明細項目) のみを表示します。

列の説明とサンプル値の詳細については、「the section called “予約の詳細”」を参照してください。

- reservation/amortizedUpfrontCostForUsage
- reservation/recurringFeeForUsage
- reservation/effectiveCost

Amazon EC2 のサイズフレキシブル予約の監視

リージョンに適用される Amazon EC2 リザーブドインスタンスは、アベイラビリティーゾーンの柔軟性とインスタンスサイズの柔軟性を提供します。アベイラビリティーゾーンの柔軟性を提供するリザーブドインスタンスでは、リージョン内の任意のアベイラビリティーゾーンでの使用に割引が提供されます。インスタンス サイズの柔軟性を提供するリザーブド インスタンスでは、そのファミリー

内のインスタンス サイズに関係なく、使用量の割引が提供されます。サイズフレキシブルリザーブドインスタンスは、最小サイズのインスタンスに最初に適用されます。詳細については、「Amazon EC2 ユーザーガイド」の「リザーブドインスタンスが適用される仕組み」を参照してください。

リザーブドインスタンスによって提供されるインスタンスサイズの柔軟性が使用状況にどのように適用されるかを理解するには、 lineItem/NormalizationFactor 列とlineItem/NormalizedUsageAmount 列を参照してください。

> **Note**
>
> インスタンスサイズの柔軟性は、デフォルトテナンシーが設定されている Linux/Unix リージョンリザーブドインスタンスでのみサポートされます。リージョンリザーブドインスタンスのインスタンスサイズ柔軟性の制限の詳細については、「Amazon EC2 ユーザーガイド」の「リージョンリザーブドインスタンスの適用方法」を参照してください。

コストと使用状況レポートでは、リザーブドインスタンスの使用量は、リザーブドインスタンスを購入したアカウントにデフォルトで適用されます。購入したアカウントが 1 時間以内に使用できないリザーブドインスタンス特典は、利用可能な利用可能なオンデマンドインスタンスの使用量に基づいて、他の連結アカウントにも適用されます。

例

特定のリージョンで `m4.xlarge` RI を 1 つ購入します。この `m4.xlarge` RI は、同じリージョン内のすべての `m4` インスタンスの使用に自動的に適用できます。次の表で、 AWS は `m4.xlarge` を 2 つの `m4.large` インスタンスに個別に適用しています。

lineItem/LineItemType	RIFee	割引使用料	割引使用料
lineItem/ProductCode	AmazonEC2	AmazonEC2	AmazonEC2
lineItem/UsageStartDate	2016-01-01T00:00:00Z	2016-01-01T00:00:00Z	2016-01-01T00:00:00Z
lineItem/UsageType	HeavyUsage:m4.xlarge	BoxUsage:m4.large	BoxUsage:m4.large
lineItem/LineItemDescription	USD 0.0618 hourly fee per Linux/UNI	Linux/UNIX (Amazon VPC), m4.large	Linux/UNIX (Amazon VPC), m4.large

	X (Amazon VPC), m4.xlarge instance	Reserved Instance applied	Reserved Instance applied
lineItem/ResourceId		i-1bd250bc	i-1df340ed
lineItem/UsageAmount		1	1
lineItem/NormalizationFactor	4	4	4
lineItem/NormalizedUsageAmount		4	4
lineItem/UnblendedRate		0	0
lineItem/UnblendedCost	46	0	0
Reservation/ReservationARN	arn:aws:ec2:us-east-1: 123456789012:reserved-instances /f8c204c1	arn:aws:ec2:us-east-1: 123456789012:reserved-instances /f8c204c1	arn:aws:ec2:us-east-1: 123456789012:reserved-instances /f8c204c1
Reservation/TotalReservedUnits	744		
Reservation/TotalReserved NormalizedUnits	5952		

この 2 つの `m4.large` 明細項目には別々の [Resource] があり、どちらも単一の `m4.xlarge` RI から割引特典を受けます。これは、使用量の [reservationARN] 値と定期的な月額料の明細項目を一致して表示されます。

RI 購入オプションの詳細については、「Amazon EC2 ユーザーガイド」の「課金の仕組み」を参照してください。

オンデマンドキャパシティー予約のモニタリング

キャパシティー予約を使用すると、特定のアベイラビリティーゾーンで任意の期間にわたってAmazon EC2 インスタンスのキャパシティーを予約できます。これにより、リージョンのリザーブドインスタンス (RI) で提供される請求割引とは別に、キャパシティー予約を作成および管理できます。請求割引の利点を享受するために、リージョンの RI をキャパシティー予約と組み合わせて使用できます。

キャパシティー予約の明細項目

AWS CUR データディクショナリで定義されている一部の列を使用して、キャパシティ予約を追跡できます。以下の列は、キャパシティー予約にも使用されます。

このセクションでは、これらの明細項目を定義し、さらにキャパシティー予約に固有の補足的な定義を示します。

コストと使用状況レポートの列の説明の詳細については、the section called “明細項目の詳細” を参照してください。

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | VWXYZ

B

lineItem/BlendedRate

[UsageType] が [Reservation] または [DedicatedRes] のキャパシティー予約の場合、[BlendedRate] は `0` です。これは、キャパシティー予約のコストが、キャパシティー予約自体ではなく、キャパシティーを提供するインスタンスに関連付けられているためです。

R

lineItem/ResourceId

コストと使用状況レポートの作成時に `lineItem/ResourceId` を含めると、[ResourceId] 列を使用してキャパシティー予約を識別して追跡できます。キャパシティー予約の ResourceId は、UnusedBox、UnusedDed、Reservation、および DedicatedRes の UsageTypes でのみキャプチャされます。

キャパシティー予約のリソース ID には常に`cr-`が含まれます。リソース ID の形式は次のとおりです。

```
arn:aws:ec2:<region>:<account id>:<capacity-reservation>/cr-0be443example1db6f
```

U

lineItem/UnblendedCost

`BlendedRate` に `UsageAmount` を乗じた値。

lineItem/UnblendedRate

[UsageType] が [Reservation] または [DedicatedRes] のキャパシティー予約の場合、[UnblendedRate] は `0` です。これは、キャパシティー予約のコストが、キャパシティー予約自体ではなく、キャパシティーを提供するインスタンスに関連付けられているためです。

lineItem/UsageAmount

使用したキャパシティー予約の量。キャパシティー予約ごとに 1 時間に複数のスロットを使用でき、1 時間にわたり、予約を使用する複数のインスタンスを実行できます。したがって、1 時間に複数のインスタンス時間を使用できます。[UsageAmount] は、明細項目で対象となるインスタンススロットの数を明細項目で対象となる時間数で掛けて計算されます。

lineItem/UsageType

使用した特定の予約の量。Amazon EC2 の場合、オプションは以下のとおりです。

lineItem/lineitemtype = BoxUsage

この `UsageType` の場合、`UsageAmount` 列は、使用したインスタンスのインスタンス時間の量です。

例えば、レポートの対象時間が 1 時間で、そのキャパシティー予約の明細項目が 10 個のインスタンスをカバーできるとします。レポートの対象時間中に 2 個のインスタンススロットを使用した場合、[BoxUsage] の [UsageAmount] は、予約して使用したインスタンス時間数を示します。この場合、2 (使用したインスタンススロットの数) に 1 時間 (レポートの対象時間) を乗算して、合計 2 となります。1 日を対象とするレポートの場合、[UsageAmount] は、2 に 24 を乗算して合計 48 になります。

DedicatedRes

[UsageType] が [DedicatedRes] の場合、[UsageAmount] 列には、予約した専用キャパシティー予約のインスタンス時間数が表示されます。

予約する

[UsageType] が [Reservation] の場合、[UsageAmount] 列には、予約したキャパシティー予約のインスタンス時間数が表示されます。

例えば、レポートの対象時間が 1 時間で、そのキャパシティー予約の明細項目が 10 個のインスタンスをカバーできる場合、[Reservation] の [UsageAmount] は、予約したインスタンススロットの数を示します。この場合、10 (使用可能なインスタンススロットの数) に 1 時間 (レポートの対象時間) を乗算して、合計 10 となります。1 日を対象とするレポートの場合、[UsageAmount] は、10 に 24 を乗算して合計 240 になります。

UnusedBox

[UsageType] が [UnusedBox] の場合、[UsageAmount] 列には、予約したが使用しなかったキャパシティー予約のインスタンス時間数が表示されます。

例えば、レポートの対象時間が 1 時間で、そのキャパシティー予約の明細項目が 10 個のインスタンスをカバーできるとします。レポートの対象時間中に 8 個のインスタンススロットを使用しなかった場合、[UnusedBox] の [UsageAmount] は、予約したが使用しなかったインスタンス時間数を示します。この場合、8 (使用していないインスタンススロットの数) に 1 時間 (レポートの対象時間) を乗算して、合計 8 となります。1 日を対象とするレポートの場合、[UsageAmount] は、8 に 24 を乗算して合計 192 になります。

UnusedDed

[UsageType] が [UnusedDed] の場合、[UsageAmount] 列には、予約したが使用しなかった専用キャパシティー予約のインスタンス時間数が表示されます。

データ転送料金について

AWS データ転送料金は、CUR AWS の lineItem/UsageType列を使用して識別できます。

> **Note**
>
> データ転送料金は、 AWS 使用するサービスと転送元の地域によって異なる場合があります。詳細な料金情報については、サービスの料金表ページを参照してください。例えば、Amazon EC2 データ転送の詳細な料金情報については、Amazon EC2 オンデマンド料金を参照してください。

AWS リージョン内のデータ転送

同じ AWS リージョン内のアベイラビリティーゾーン間のデータ転送の UsageType は ですRegion-DataTransfer-Regional-Bytes。例えば、USE2-DataTransfer-Regional-Bytes使用タイプによって、米国東部 (オハイオ) リージョンのアベイラビリティーゾーン間のデータ転送の料金が決まります。

特定のリソースについては、 AWS リージョン内のデータ転送におけるインバウンドトラフィックとアウトバウンドトラフィックの両方に対して課金されます。つまり、従量制リソースごとに、データ転送ごとに 2 DataTransfer-Regional-Bytes つの明細項目が表示されます。一部のサービスではリージョン内のトラフィックが無料で発生するため、詳細についてはサービスの料金ページを確認してください。

AWS リージョン間のデータ転送

異なる AWS リージョン間のデータ転送には、次の使用タイプを使用できます。

- *Source Region-Destination Region*-AWS-In-Bytes: 別の特定の AWS リージョンから送信先リージョンへの受信データ転送を測定します。
- *Source Region-Destination Region*-AWS-Out-Bytes: ソースリージョンから別の特定の AWS リージョンへの送信データ転送を測定します。
- *Source Region*-AWS-In-Bytes: この使用タイプは、トラフィックが VPC ピアリングを介して流れるときに表示されます。
- *Source Region*-AWS-Out-Bytes: この使用タイプは、トラフィックが VPC ピアリングを介して流れるときに表示されます。

リソースごとに、 AWS リージョン間のデータ転送はレポート内の 2 つの明細項目に対応します。

- 転送先のリージョンに転送されるデータの 1 つの明細項目
- ソースリージョンから転送されるデータの明細項目。

転送先のリージョンに転送されたデータには料金はかかりません。データ転送料金は、転送元リージョンから転送されるデータによって決まります。

たとえば、USE2リージョンからAPS3リージョンへのデータ転送には、APS3-USE2-AWS-In-Bytes明細項目とUSE2-APS3-AWS-Out-Bytes明細項目の両方が含まれます。APS3-USE2-AWS-In-Bytesこの明細項目には対応する料金はありません。USE2-APS3-AWS-Out-Bytesデータ転送料金は明細項目に関連付けられています。

インターネットへのデータ転送。

からインターネット AWS へのデータ転送の UsageType は です*Region*-DataTransfer-Out-Bytes。たとえば、USE2-DataTransfer-Out-Bytes使用タイプによって、USE2リージョンからインターネットへのデータ転送の料金が決まります。

インターネットからへのデータ転送には料金はかかりません AWS。

> **Note**
>
> リージョンプレフィックスが付いていないデータ転送使用タイプ (DataTransfer-Regional-Bytesやなど) はDataTransfer-Out-Bytes、米国東部 (バージニア北部) リージョンからのデータ転送を表します。

AWS Direct Connect トラフィック

AWS Direct Connect パブリック仮想インターフェイスを介したデータ転送には、 DataXfer-Inまたは で終わる使用タイプがありますDataXfer-Out。

AWS Direct Connect プライベートまたはトランジット仮想インターフェイスを介したデータ転送には、 DataXfer-In:dc.3 または で終わる使用タイプがありますDataXfer-Out:dc.3。

S3 Transfer Acceleration

S3 転送アクセラレーションを使用した Amazon S3 データ転送には、ABytes次のような使用タイプがあります。

- Amazon S3 と Amazon EC2 の間: C3DataTransfer-In-ABytes または C3DataTransfer-Out-ABytes で終わる使用タイプ
- Amazon S3 とインターネット間: DataTransfer-In-ABytesまたは DataTransfer-Out-ABytes で終わる使用タイプ
- Amazon S3 とCloudFront の間: CloudFront-In-ABytes または CloudFront-Out-ABytes で終わる使用タイプ
- 異なる AWS リージョンの Amazon S3 バケット間: の使用タイプ *Source Region*-*Destination Region*-AWS-Out-ABytes

CloudFront トラフィック

CloudFront データ転送の使用タイプは、`Region-DataTransfer-Out-Bytes`*Region*`-DataTransfer-Out-OBytes`または製品コードと組み合わせたものです。`AmazonCloudFront`使用タイプのリージョンプレフィックスは、データ転送に使用される CloudFront Edge ロケーションを指します。例えば、`AP-DataTransfer-Out-Bytes` 使用タイプによって、AP リージョンからインターネットへのデータ転送の料金が決まります。

> **Tip**
>
> lineItem/ProductCodeこの列を使用して、CloudFront のデータ転送とインターネットへのデータ転送を区別します。これらのデータ転送タイプの使用タイプは似ています。

分割コスト配分データを理解する

コストと使用状況レポート (AWS CUR) を使用して、Amazon ECS と Amazon EKS コンテナのコストを追跡できます。分割コスト配分データを使用すると、コンテナワークロードが共有のコンピューティングリソースとメモリリソースをどのように消費しているかに基づいて、コンテナコストを個々のビジネスユニットやチームに割り当てることができます。分割コスト配分データは、新しいコンテナレベルのリソース (ECS タスクと Kubernetes ポッド) のコストと使用状況データを AWS CUR に導入します。以前は、 AWS CUR は EC2 インスタンスレベルでのみコストをサポートしていました。分割コスト配分データでは、各コンテナの EC2 インスタンスのリソース消費量を調べてコンテナレベルのコストを生成し、インスタンスの償却コストと、インスタンス上で実行されたコンテナが消費する CPU およびメモリリソースの割合に基づいてコストを生成します。

分割コスト配分データでは、CUR 内のコンテナ化されたリソース ID (ECS タスクと Kubernetes ポッド) AWS ごとに新しい使用状況レコードと新しいコストメトリクス列が導入されます。詳細については、「明細の分割の詳細」を参照してください。

分割コスト配分データを CUR AWS に含めると、CPU とメモリのコストを反映するために、ECS タスクと Kubernetes ポッドごとに 1 時間あたり 2 つの新しい使用状況レコードが追加されます。1 日あたりの CUR AWS の新しい明細項目の数を見積もるには、次の式を使用します。

ECS の場合: `(number of tasks * average task lifetime * 2) * 24`

EKS の場合: `(number of pods * average pod lifetime * 2) * 24`

例えば、10 個の EC2 インスタンスからなるクラスターで 1 時間に 1,000 件のポッドが実行されており、そのポッドの有効期間が 1 時間未満の場合、

```
(1000 * 1 * 2) * 24 = 48,000 new usage records in AWS CUR
```

> **Note**
>
> ECS の場合: AWS コスト配分タグについては、コストと使用状況レポートに Amazon ECS 管理タグまたはユーザー追加タグを使用できます。これらのタグは、すべての新しい ECS 分割コスト配分データ使用レコードに適用されます。詳細については、[課金用の ECS リソースのタグ付け]を参照してください。
> EKS の場合: 分割コスト配分データは、一部の Kubernetes 属性に新しいコスト配分タグを作成します。これらのタグには、`aws:eks:cluster-name`、`aws:eks:deployment`、`aws:eks:namespace`、`aws:eks:node`、`aws:eks:workload-name`、`aws:eks:workload-type` が含まれます。
>
> - `aws:eks:cluster-name`、`aws:eks:namespace`、`aws:eks:node` には、クラスター、名前空間、ノードの名前が遡及的に入力されます。
> - `aws:eks:workload-type` は、ポッドを管理するワークロードが 1 つだけあり、組み込みワークロードの 1 つである場合にのみ入力されます。ワークロードタイプには、`ReplicaSet`、`StatefulSet`、`Job`、`DaemonSet`、または が含まれ`ReplicationController`、ワークロードの名前`aws:eks:workload-name`が含まれます。詳細については、「Kubernetes ドキュメント」の「ワークロード」を参照してください。
> - `aws:eks:deployment` はワークロードタイプ `ReplicaSet` にのみ入力されます。`ReplicaSet` を作成するのはデプロイです。
>
> これらのタグは、すべての新しい EKS 分割コスト配分データ使用レコードに適用されます。これらのタグは、デフォルトでコスト配分が有効になっています。`aws:eks:cluster-name` タグを以前に使用および無効にした場合、分割コスト配分データはこの設定を保持し、タグを有効にしません。これは、[コスト配分タグ] コンソールページから有効にできます。

分割コスト配分データを有効にする

> **Note**
>
> 分割コスト配分データは Cost Explorer では使用できません。従来のコストと使用状況レポート (CUR)、およびデータエクスポートを備えたコストと使用状況レポート 2.0 (CUR 2.0) で使用できます。

コスト管理の設定でコスト配分データを分割するよう選択することが前提条件です。

コスト配分データの分割をオプトインするには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインで [コスト管理の詳細設定] を選択します。
3. [全般] の [コスト配分データの分割] セクションで、次のいずれかを選択します。
   - Amazon ECS にのみオプトインする場合は [Amazon Elastic Container Service (Amazon ECS)]。
   - Amazon EKS にのみオプトインする場合は [Amazon Elastic Kubernetes Service (Amazon EKS)]。Amazon EKS の場合は、次のいずれかを選択します。
     - リソースリクエスト: Amazon EC2 は Kubernetes ポッド CPU とメモリリソースによってのみ割り当てられます。これにより、アプリケーションチームは必要なもののみをプロビジョニングできます。
     - Amazon Managed Service for Prometheus: Kubernetes ポッドの CPU とメモリのリソース要求と実際の使用率のいずれか高い方に基づいて Amazon EC2 のコストが割り当てられます。これにより、各アプリケーションチームが使用した分だけ支払うことになります。Amazon Managed Service for Prometheus のセットアップの詳細については、「Amazon Managed Service for Prometheus ユーザーガイド」の「Setting up」を参照してください。

       前提条件: ですべての機能を有効にする必要があります AWS Organizations。詳細については、「Organizations ユーザーガイド」の「組織内のすべての機能の有効化」を参照してください。

- Amazon CloudWatch Container Insights: これにより、共有 EC2 インスタンスを使用して複数のアプリケーションコンテナを実行しているクラスターのコストをより詳細に可視化できるため、EKS クラスターの共有コストのコスト配分が向上します。

> **Note**
>
> - 通常のアカウントと支払者アカウントのみが AWS Cost Management 設定にアクセスでき、コスト配分データの分割をオプトインできます。オプトインすると、メンバーアカウントはコストと使用状況レポートのデータを表示できます。
> - リソースリクエストを選択すると、メモリと CPU リクエストで設定されたポッドのみが分割コスト配分データに使用されます。使用をリクエストしていないポッドには、分割コストデータが表示されません。
> - Amazon Managed Service for Prometheus を選択した場合は、 AWS Organizations のすべての機能を有効にする必要があります。詳細については、「組織内のすべての機能の有効化」を参照してください。さらに、分割コスト配分データでは、サービスにリンクされたロールで使用する新しいポリシーが作成されます。これで、分割コスト配分データによって使用または管理される AWS サービスやリソースにアクセスできるようになります。

オプトインすると、レポート作成のステップ 1 でコンテナレベルのリソースのコストと使用状況データをレポートに含めるか、後でレポートの詳細を編集してレポートに含めるかを選択できます。

コストと使用状況のデータをレポートに含めるには

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. 新しいレポートを作成するか、既存のレポートを編集するかにかかわらず、[レポートの詳細を指定] ページの [レポートコンテンツ] で [コスト配分データの分割] を選択します。

> **Note**
>
> CUR API または AWS Command Line Interface (CLI) AWS を使用して、分割コスト配分データの設定を管理することもできます。

分割コスト配分データにより、一括決済ファミリー全体 (支払いアカウントと連結アカウント) のすべての Amazon ECS と Amazon EKS のコンテナオブジェクトのコストを可視化できます。分割コスト配分データを有効にすると、タスクとコンテナが自動的にスキャンされます。コンテナワークロードのテレメトリ使用状況データを取り込み、当月の詳細なコストデータを準備します。

> **Note**
>
> CUR にデータが表示されるまでに最大 24 AWS 時間かかる場合があります。

Billing and Cost Management 管理コンソールページへのアクセス権の管理の詳細については、「アクセス権限の管理の概要」を参照してください。

Cost Explorer AWS Cost Management の設定とアクセスの制御については、Cost Explorer へのアクセスの制御」を参照してください。 Cost Explorer

分割コスト配分データの例

次の例の目的は、個々の Amazon ECS サービス、Amazon ECS クラスター内のタスク、Amazon EKS クラスター内の Kubernetes 名前空間とポッドのコストを計算することによって、分割コスト割り当てデータがどのように計算されるかを示すことです。例全体で使用されている料金は、説明のみを目的としています。

> **Note**
>
> この例では、Amazon EKS クラスターで実行されている Kubernetes 名前空間とポッドを示しています。その後、Amazon ECS クラスターで実行されている Amazon ECS サービスとタスクに同じコストモデルを適用できます。

1 時間あたりの使用量は以下のとおりです。

- 2 つの名前空間と 4 つのポッドで構成される単一インスタンス (m5.xlarge) 共有クラスターで、1 時間稼働します。
- インスタンス構成は 4 vCPU と 16 GB です。
- インスタンスの償却コストは 1 時間あたり USD です。

分割コスト配分データでは、CPU とメモリの相対的な単位重量を 9:1 の比率で計算しています。これは、の vCPU 1 時間あたりの料金と 1 時間あたり GB あたりの料金から算出されます。AWS Fargate

ステップ 1: CPU とメモリの単価を計算する

```
Unit-cost-per-resource = Hourly-instance-cost/((Memory-weight * Memory-available) + (CPU-weight * CPU-available))
```

= 1ドル/ ((1 * 16GB) + (9 * 4vCPU)) = 0.02 ドル

```
Cost-per-vCPU-hour = CPU-weight * Unit-cost-per-resource
```

= 9 * 0.02 ドル = 0.17 ドル

```
Cost-per-GB-hour = Memory-weight * Unit-cost-per-resource
```

= 1 * 0.02 ドル = 0.02 ドル

Instance	Instance type	vCPU-available	Memory-available	Amortized-cost-per-hour	Cost-per-vCPU-hour	Cost-per-GB-hour
Instance1	m5.xlarge	4	16	1 USD	0.17 USD	0.02 USD

ステップ 2: 割り当てられた容量とインスタンスの未使用容量を計算する

- 割り当て済み容量:親 EC2 インスタンスから Kubernetes ポッドに割り当てられたメモリと vCPU。使用済み容量とリザーブドキャパシティの最大値として定義されます。

> **Note**
> メモリまたは vCPU の使用状況データがない場合は、代わりに予約データが使用されます。詳細については、「Amazon ECS 使用状況レポート」または「Amazon EKS コストモニタリング」を参照してください。

- インスタンスの未使用容量: vCPU とメモリの未使用容量。

```
Pod1-Allocated-vCPU = Max (1 vCPU, 0.1 vCPU) = 1 vCPU
```

`Pod1-Allocated-memory = Max (4 GB, 3 GB)` = 4 GB

`Instance-Unused-vCPU = Max (CPU-available - SUM(Allocated-vCPU), 0)`= 最大値 (4 — 4.9、0) = 0

`Instance-Unused-memory = Max (Memory-available - SUM(Allocated-memory), 0)`= 最大 (16 — 14、0) = 2 GB

この例では、インスタンスの CPU がサブスクリプションを超えています。これは、予約されたものよりも多くの vCPU を使用した Pod2 に起因します。

Pod name	Namespace	Reserved-vCPU	Used-vCPU	Allocated-vCPU	Reserved-memory	Used-memory	Allocated-memory
Pod1	Namespace 1	1	0.1	1	4	3	4
Pod2	Namespace 2	1	1.9	1.9	4	6	6
Pod3	Namespace 1	1	0.5	1	2	2	2
Pod4	Namespace 2	1	0.5	1	2	2	2
Unused	Unused			0			2
				4.9			16

ステップ 3: 分割使用率を計算する

- 分割使用率: EC2 インスタンスで使用可能な CPU またはメモリ全体に対する Kubernetes ポッドが使用している CPU またはメモリの割合。
- 未使用率: EC2 インスタンスで使用されている CPU またはメモリ全体の (つまり、インスタンスの未使用の CPU またはメモリは考慮されていない) に対する Kubernetes ポッドが使用している CPU またはメモリの割合。

`Pod1-vCPU-split-usage-ratio = Allocated-vCPU / Total-vCPU`

= 1 vCPU/4.9vCPU = 0.204

`Pod1-Memory-split-usage-ratio = Allocated-GB / Total-GB`

= 4 ギガバイト/ 16ギガバイト = 0.250

`Pod1-vCPU-unused-ratio = Pod1-vCPU-split-usage-ratio / (Total-CPU-split-usage-ratio – Instance-unused-CPU)` (インスタンス未使用 CPU が 0 の場合は 0 に設定)

= 0 (インスタンス未使用 CPU は 0 なので)

`Pod1-Memory-unused-ratio = Pod1-Memory-split-usage-ratio / (Total-Memory-split-usage-ratio – Instance-unused-memory)` (インスタンス未使用メモリが 0 の場合は 0 に設定)

= 0.250/(1-0.125) = 0.286

Pod name	Namespace	vCPU-split-usage-ratio	vCPU-unused-ratio	Memory-split-usage-ratio	Memory-unused-ratio
Pod1	Namespace1	0.204	0	0.250	0.286
Pod2	Namespace2	0.388	0	0.375	0.429
Pod3	Namespace1	0.204	0	0.125	0.143
Pod4	Namespace2	0.204	0	0.125	0.143
Unused	Unused	0		0.125	
		1		1	

ステップ 4: 分割コストと未使用コストの計算

- 分割コスト: Kubernetes ポッドによって割り当てられた CPU とメモリの使用量に基づく EC2 インスタンスコストの従量課金制コスト配分。
- 未使用インスタンスコスト: インスタンス上の未使用の CPU またはメモリリソースのコスト。

`Pod1-Split-cost = (Pod1-vCPU-split-usage-ratio * vCPU-available * Cost-per-vCPU-hour) + (Pod1-Memory-split-usage-ratio * Memory-available * Cost-per-GB-hour)`

= (0.204 * 4 vCPU * 0.17 ドル) + (0.25 * 16GB * 0.02 ドル) = 0.22 ドル

`Pod1-Unused-cost = (Pod1-vCPU-unused-ratio * Instance-vCPU-unused-ratio * vCPU-available * Cost-per-VCPU-hour) + (Pod1-Memory-unused-ratio * Instance-Memory-unused ratio * Memory-available * Cost-per-GB-hour)`

= (0 * 0 * 4 * 0.17 ドル) + (0.286 * 0.125 * 16 * 0.02 ドル) = 0.01 ドル

`Pod1-Total-split-cost = Pod1-Split-cost + Pod1-Unused-cost`

= 0.23 ドル

Pod name	Namespace	Split-cost	Unused-cost	Total-split-cost
Pod1	Namespace1	0.22 USD	0.01 USD	0.23 USD
Pod2	Namespace2	0.38 USD	0.02 USD	0.40 USD
Pod3	Namespace1	0.18 USD	0.01 USD	0.19 USD
Pod4	Namespace2	0.18 USD	0.01 USD	0.19 USD
Unused	Unused	0.04 USD		
		1 USD	0.04 USD	1 USD

サービスのコストは、各名前空間に関連するポッドのコストの合計です。

Namespace1 の合計コスト = $0.23 + $0.19 = $0.42

Namespace2 の合計コスト = $0.40 + $0.19 = $0.59

サンプル AWS CUR

請求期間中の EC2 インスタンスの使用量全体をカバーするSavings Plan をお持ちの場合、償却コストは savingsPlan/SavingsPlanEffectiveCost を使用して計算されます。

lineItem/ ResourceID	lineItem/ lineItemType	lineItem/ UsageType	lineItem/ UnblendedCost	lineItem/ NetUnblendedCost	savingsPlan/ SavingsPlanEffectiveCost	savingsPlan/ NetSavingsPlanEffectiveCost	splitLineItem/ ParentResourceId	splitLineItem/ SplitUsage	splitLineItem/ SplitCost	splitLineItem/ NetSplitCost	splitLineItem/ UnusedCost	splitLineItem/ NetUnusedCost
i-12345	SavingsPlan CoveredUsage	BoxUsage: m5.xlarge	1.5	1.4	1	0.8						
EC2-Pod1	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod1	Usage	EKS-EC2-GB-Hours					i-12345	4	0.08	0.06	0.01	0.01
EC2-Pod2	Usage	EKS-EC2-vCPU-Hours					i-12345	1.9	0.27	0.21	0	0
EC2-Pod2	Usage	EKS-EC2-GB-Hours					i-12345	6	0.12	0.09	0.02	0.01
EC2-Pod3	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod3	Usage	EKS-EC2-GB-Hours					i-12345	2	0.04	0.03	0.01	0
EC2-Pod4	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod4	Usage	EKS-EC2-GB-Hours					i-12345	2	0.04	0.03	0.01	0

請求期間内の EC2 インスタンスの一部使用分をカバーするSavings Plan があり、残りの EC2 インスタンスの使用量がオンデマンド料金で請求される場合、EC2 インスタンスの償却費用は (SavingsPlanCoveredUsage) + savingsPlan/SavingsPlanEffectiveCost (オンデマンド使用の場合) を使用して計算されます。lineItem/UnblendedCost

lineItem/ ResourceID	lineItem/ lineItemType	lineItem/ UsageType	lineItem/ UnblendedCost	lineItem/ NetUnblendedCost	savingsPlan/ SavingsPlanEffectiveCost	savingsPlan/ NetSavingsPlanEffectiveCost	splitLineItem/ ParentResourceId	splitLineItem/ SplitUsage	splitLineItem/ SplitCost	splitLineItem/ NetSplitCost	splitLineItem/ UnusedCost	splitLineItem/ NetUnusedCost
i-12345	SavingsPlan CoveredUsage	BoxUsage: m5.xlarge	1.2	0.9	0.8	0.65						
i-12345	Usage	BoxUsage: m5.xlarge	0.2	0.15								
EC2-Pod1	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod1	Usage	EKS-EC2-GB-Hours					i-12345	4	0.08	0.06	0.01	0.01
EC2-Pod2	Usage	EKS-EC2-vCPU-Hours					i-12345	1.9	0.27	0.21	0	0
EC2-Pod2	Usage	EKS-EC2-GB-Hours					i-12345	6	0.12	0.09	0.02	0.01
EC2-Pod3	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod3	Usage	EKS-EC2-GB-Hours					i-12345	2	0.04	0.03	0.01	0
EC2-Pod4	Usage	EKS-EC2-vCPU-Hours					i-12345	1	0.14	0.11	0	0
EC2-Pod4	Usage	EKS-EC2-GB-Hours					i-12345	2	0.04	0.03	0.01	0

Amazon Managed Service for Prometheus で分割コスト配分データを使用する

Amazon EKS のコストデータを分割するには、メモリや CPU 使用量などのメトリクスをクラスターから収集して保存する必要があります。この目的には、Amazon Managed Service for Prometheus を使用できます。

コスト配分データの分割をオプトインし、Amazon Managed Service for Prometheus ワークスペースが 2 つの必須メトリクス (`container_cpu_usage_seconds_total` および `container_memory_working_set_bytes`) の受信を開始すると、分割コスト配分データはメトリクスを認識し、自動的に使用します。

> **Note**
>
> 2 つの必須メトリクス (`container_cpu_usage_seconds_total` と `container_memory_working_set_bytes`) は、デフォルトの Prometheus スクレイプ設定と、 AWS マネージドコレクターで提供されるデフォルト設定にあります。ただし、これ

らの設定をカスタマイズする場合は、`container_cpu_usage_seconds_total` および `container_memory_working_set_bytes` メトリクスから、`name`、`namespace`、`pod` のラベルを再ラベル付けしたり、変更したり、削除したりしないでください。これらのラベルの再ラベル付け、変更、修正、または削除を行うと、メトリクスの取り込みに影響する可能性があります。

Amazon Managed Service for Prometheus を使用して、1 つのリージョンで 1 つの使用状況アカウントから EKS メトリクスを収集できます。Amazon Managed Service for Prometheus ワークスペースは、そのアカウントとリージョンにある必要があります。コストをモニタリングする使用状況アカウントとリージョンごとに、Amazon Managed Service for Prometheus インスタンスが 1 つ必要です。Amazon Managed Service for Prometheus ワークスペース内の複数のクラスターのメトリクスは、同じ使用状況アカウントとリージョンにある限り収集できます。

以下のセクションでは、EKS クラスターから Amazon Managed Service for Prometheus ワークスペースに正しいメトリクスを送信する方法について説明します。

前提条件

分割コスト配分データで Amazon Managed Service for Prometheus を使用するための前提条件:

- AWS Billing and Cost Management コンソールで分割コスト配分データを有効にする必要があります。詳細については、「Enabling split cost allocation data」を参照してください。コスト配分データの分割にオプトインすると、各使用アカウントにサービスにリンクされたロールが作成され、そのアカウントの Amazon EKS クラスターメトリクスについて Amazon Managed Service for Prometheus にクエリが実行されます。詳細については、「分割コスト配分データのサービスにリンクされたロール」を参照してください。
- 分割コスト配分データを追跡する EKS クラスターが必要です。新しいクラスターを作成するか、既存のクラスターを使用できます。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS クラスターの作成」を参照してください。

> **Note**
>
> 後の手順で使用するために、`EKS cluster ARN`、`security group IDs`、さらに少なくとも 2 つの `subnet IDs` (異なるアベイラビリティーゾーン内) が必要になります。(任意) EKS クラスターの認証モードを `API` または `API_AND_CONFIG_MAP` に設定します。

- EKS クラスターと同じアカウントとリージョンに Amazon Managed Service for Prometheus インスタンスが必要です。まだ持っていない場合は、作成する必要があります。Amazon Managed Service for Prometheus インスタンスを作成する方法については、「Amazon Managed Service for Prometheus ユーザーガイド」の「Create a workspace」を参照してください。

> **Note**
>
> 後の手順で使用するために `Amazon Managed Service for Prometheus workspace` ARN が必要です。

EKS メトリクスを Amazon Managed Service for Prometheus に転送する

EKS クラスターと Amazon Managed Service for Prometheus インスタンスがあれば、メトリクスをクラスターからインスタンスに転送できます。メトリクスは 2 つの方法で送信できます。

- オプション 1: AWS マネージドコレクターを使用します。これは、EKS クラスターから Amazon Managed Service for Prometheus にメトリクスを送信する最も簡単な方法です。ただし、最大 30 秒ごとにメトリックをスクレイピングするだけという制限があります。
- オプション 2: 独自の Prometheus エージェントを作成します。この場合、スクレイピング構成をより細かく制御できますが、エージェントを作成した後に管理する必要があります。

オプション 1: AWS マネージドコレクターの使用

AWS マネージドコレクター (スクレイパー) を使用することは、EKS クラスターから Amazon Managed Service for Prometheus インスタンスにメトリクスを送信する最も簡単な方法です。次の手順では、 AWS マネージドコレクターを作成する手順を示します。詳細については、「Amazon Managed Service for Prometheus ユーザーガイド」の「AWS managed collectors」を参照してください。

> **Note**
>
> AWS マネージドコレクターの最小スクレイプ間隔は 30 秒です。有効期間の短いポッドがある場合は、スクレイパー間隔を 15 秒に設定することをお勧めします。15 秒のスクレイパー間隔を使用するには、オプション 2 を使用して独自の Prometheus エージェントを作成します。

AWS マネージドコレクターを作成するには、次の 3 つのステップがあります。

1. スクレイパー設定を作成します。
2. スクレイパーを作成します。
3. EKS クラスターを、スクレイパーがメトリクスにアクセスできるように設定します。

ステップ 1: 設定を作成する

スクレイパーを作成するには、スクレイパー設定が必要です。デフォルト設定を使用するか、独自の設定を作成できます。スクレイパー設定を取得する 3 つの方法は次のとおりです。

- CLI AWS を使用してデフォルト設定を取得するには、以下を呼び出します。

```
aws amp get-default-scraper-configuration
```

- 独自の設定を作成します。詳細については、「Amazon Managed Service for Prometheus ユーザーガイド」の「Scraper configuration」を参照してください。
- 「Amazon Managed Service for Prometheus ユーザーガイド」の同じ「Scraper configuration」の手順で提供されているサンプル設定をコピーします。

スクレイパー設定を編集して、スクレイプ間隔を変更したり、スクレイプされたメトリクスをフィルタリングしたりできます。

スクレイピングされるメトリクスをフィルタリングして、分割コスト配分データに必要な 2 つのメトリクスを含めるには、次のスクレイパー設定を使用します。

```
global:
   scrape_interval: 30s
   #external_labels:
     #clusterArn: <REPLACE_ME>
scrape_configs:
  - job_name: kubernetes-nodes-cadvisor
    scrape_interval: 30s
    scrape_timeout: 10s
    scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    kubernetes_sd_configs:
```

```
    - role: node
    relabel_configs:
    - regex: (.+)
      replacement: /api/v1/nodes/$1/proxy/metrics/cadvisor
      source_labels:
      - __meta_kubernetes_node_name
      target_label: __metrics_path__
    - replacement: kubernetes.default.svc:443
      target_label: __address__
    metric_relabel_configs:
    - source_labels: [__name__]
      regex: 'container_cpu_usage_seconds_total|container_memory_working_set_bytes'
      action: keep
```

スクレイパー設定が完了したら、ステップ 2 で使用するために base64 でエンコードする必要があります。設定はテキスト YAML ファイルです。ファイルをエンコードするには、https://www.base64encode.org/ などのウェブサイトを使用します。

ステップ 2: スクレイパーを作成する

設定ファイルを作成したら、スクレイパーを作成する必要があります。前提条件セクションで説明されている変数に基づいて、次の AWS CLI コマンドを使用してスクレイパーを作成します。*<EKS-CLUSTER-ARN>*、*<SG-SECURITY-GROUP-ID>*、*<SUBNET-ID>* の各フィールドに EKS クラスターの情報を使用し、*<BASE64-CONFIGURATION-BLOB>* を前のステップで作成したスクレイパー設定に置き換え、*<AMP_WORKSPACE_ARN>* を Amazon Managed Service for Prometheus ワークスペース ARN に置き換える必要があります。

```
aws amp create-scraper \
--source eksConfiguration="{clusterArn=<EKS-CLUSTER-ARN>,securityGroupIds=[<SG-SECURITY-GROUP-ID>],subnetIds=[<SUBNET-ID>]}" \
--scrape-configuration configurationBlob=<BASE64-CONFIGURATION-BLOB> \
--destination ampConfiguration={workspaceArn="<AMP_WORKSPACE_ARN>"}
```

ステップ 3 で使用するために返される `scraperId` を書き留めます。

ステップ 3: EKS クラスターを、スクレイパーがメトリクスにアクセスできるように設定する

EKS クラスターの認証モードが `API` または `API_AND_CONFIG_MAP` のいずれかに設定されている場合、スクレイパーは自動的に正しいクラスター内アクセスポリシーを持ち、スクレイパーはクラスターにアクセスできます。これ以上の設定は必要なく、メトリクスは Amazon Managed Service for Prometheus に流れるはずです。

EKS クラスターの認証モードが `API` または `API_AND_CONFIG_MAP` に設定されていない場合は、スクレイパーが ClusterRole および ClusterRoleBinding を介してメトリクスにアクセスできるようにクラスターを手動で設定する必要があります。これらのアクセス許可を有効にする方法については、「Amazon Managed Service for Prometheus ユーザーガイド」の「Manually configuring an EKS cluster for scraper access」を参照してください。

スクレイパーがアクティブになったら、両方のメトリクス (`container_cpu_usage_seconds_total` と `container_memory_working_set_bytes`) が Amazon Managed Service for Prometheus ワークスペースにプッシュされていることを確認します。

```
awscurl --service="aps" --region="<REGION>" "https://aps-
workspaces.<REGION>.amazonaws.com/workspaces/<WorkSpace_ID>/api/v1/label/__name__/
values"
```

出力:

```
{
"status": "success",
"data": [
"container_cpu_usage_seconds_total",
"container_memory_working_set_bytes",
"scrape_duration_seconds",
"scrape_samples_post_metric_relabeling",
"scrape_samples_scraped",
"scrape_series_added",
"up"
]
}
```

オプション 2: 独自の Prometheus エージェントを作成する

AWS マネージドコレクターを使用できない場合、または独自の Prometheus サーバーがすでにある場合は、独自の Prometheus インスタンスをエージェントとして使用して EKS クラスターからメトリクスをスクレイプし、Amazon Managed Service for Prometheus に送信できます。

独自の Prometheus インスタンスをエージェントとして使用する方法の詳細については、「Amazon Managed Service for Prometheus ユーザーガイド」の「Using a Prometheus instance as a collector」を参照してください。

以下は、Prometheus サーバースクレイプ間隔と分割コスト配分データに必要なコンテナメトリクスを含む Prometheus スクレイプ設定の例です。ポッドの有効期間が短い場合は、デフォルトの Prometheus サーバースクレイプ間隔を 30 秒から 15 秒に短縮することをお勧めします。この場合、Prometheus サーバーのメモリ使用率が高くなる可能性があります。

```
global:
   scrape_interval: 30s
   #external_labels:
     #clusterArn: <REPLACE_ME>
scrape_configs:
  - job_name: kubernetes-nodes-cadvisor
    scrape_interval: 30s
    scrape_timeout: 10s
    scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    kubernetes_sd_configs:
    - role: node
    relabel_configs:
    - regex: (.+)
      replacement: /api/v1/nodes/$1/proxy/metrics/cadvisor
      source_labels:
      - __meta_kubernetes_node_name
      target_label: __metrics_path__
    - replacement: kubernetes.default.svc:443
      target_label: __address__
    metric_relabel_configs:
    - source_labels: [__name__]
      regex: 'container_cpu_usage_seconds_total|container_memory_working_set_bytes'
      action: keep
```

「Amazon Managed Service for Prometheus ユーザーガイド」で、「Set up ingestion from a new Prometheus server using Helm」に従った場合は、スクレイプ設定を更新できます。

スクレイプ設定を更新するには

1. ガイドの `my_prometheus_values_yaml` を編集し、サンプルスクレイプ設定を `server` ブロックに含めます。
2. 「Amazon Managed Service for Prometheus ユーザーガイド」の `prometheus-chart-name` および `prometheus-namespace` を使用して、次のコマンドを実行します。

```
helm upgrade prometheus-chart-name prometheus-community/prometheus -n prometheus-
namespace -f my_prometheus_values_yaml
```

`scrape_interval` の詳細またはグローバル以外の scrape_interval の使用方法については、Prometheus の「<scrape_config>」を参照してください。

または、Prometheus Receiver、Prometheus Remote Write Exporter、および AWS Sigv4 Authentication Extension を持つ AWS Distro for OpenTelemetry コレクターを使用して、Amazon Managed Service for Prometheus へのリモート書き込みアクセスを実現することもできます。

> **Note**
>
> Prometheus エージェントを設定したら、 AWS マネージドコレクターとは異なり、エージェントを最新の状態に保ち、メトリクスを収集するために実行する必要があります。

Amazon Managed Service for Prometheus のコストの見積もり

AWS 料金計算ツールを使用して、分割コスト配分データに Amazon Managed Service for Prometheus を使用するコストを見積もることができます。

見積りに Amazon Managed Service for Prometheus を設定するには

1. https://calculator.aws/#/ で AWS 料金計算ツールを開きます。
2. [Create estimate (見積りの作成)] を選択します。
3. [サービスの追加]ページで、検索フィールドに「Amazon Managed Service for Prometheus」と入力し、[設定] を選択します。
4. [Description] (説明) の欄に見積りの説明を入力します。
5. [Region] (リージョン) を選択します。
6. [インフラストラクチャの詳細を使用してコストを計算する] を選択します。このオプションを使用すると、現在または提案されているインフラストラクチャの設定に基づいて、取り込み、ストレージ、クエリのサンプルコストを見積もることができます。
7. [EC2 インスタンスの数] には、一括決済ファミリー全体 (すべてのアカウントとリージョンを含む) のすべてのクラスターの EC2 インスタンスの合計数を入力します。を使用する場合は AWS Fargate、EC2 インスタンス数のプロキシとして Fargate タスクの数を使用します。

8. 分割コスト配分データには、`container_cpu_usage_seconds_total` と `container_memory_working_set_bytes` の 2 つのメトリクスが必要です。[EC2 インスタンスあたりの Prometheus メトリクス] には、「2」と入力します。
9. 分割コスト配分データは、スクレイプ間隔が 15 秒であることを示しています。[メトリクス収集間隔 (秒)] には、「15」と入力します。別の間隔 (30 秒など) を使用した場合は、設定した間隔に変更します。
10. 分割コスト配分データは、他のパラメータに特定の要件を課さないため、ビジネス要件に従って残りの入力パラメータに適切な値を入力します。
11. [サービスを保存して追加] を選択します。

Amazon CloudWatch Container Insights での分割コスト配分データの使用

Amazon EKS のコストデータを分割するには、メモリや CPU 使用量などのメトリクスをクラスターから収集して保存する必要があります。Amazon CloudWatch Container Insights はこの目的に使用できます。

コスト配分データの分割をオプトインし、EKS クラスターで EKS オブザーバビリティアドオンを使用して CloudWatch エージェントを設定すると、分割コスト配分データは 2 つの必要なメトリクス(`pod_cpu_usage_total`と `pod_memory_working_set`) `ContainerInsights`を名前空間で受信し始め、自動的に使用されます。EKS のコンテナメトリクスの完全なセットを表示するには、「Amazon CloudWatch ユーザーガイド」の「Amazon EKS と Kubernetes Container Insights のメトリクス」を参照してください。 Amazon CloudWatch

以下のセクションでは、EKS クラスターから正しいメトリクスを送信してコスト配分データを分割する方法について説明します。

前提条件

分割コスト配分データで Amazon CloudWatch Container Insights を使用するための前提条件：

- AWS Billing and Cost Management コンソールで分割コスト配分データを有効にする必要があります。詳細については、「Enabling split cost allocation data」を参照してください。
- 分割コスト配分データを追跡する EKS クラスターが必要です。新しいクラスターを作成するか、既存のクラスターを使用できます。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS クラスターの作成」を参照してください。

EKS メトリクスを転送するように Amazon CloudWatch Container Insights を設定する

EKS メトリクスを転送するには、CloudWatch エージェントをセットアップして設定する必要があります。Amazon CloudWatch Observability EKS アドオンまたは Amazon CloudWatch Observability Helm チャートを使用して、EKS クラスターに CloudWatch エージェントと Fluent-bit エージェントをインストールできます。CloudWatch エージェントをインストールしてセットアップする方法の詳細については、Amazon CloudWatch ユーザーガイド」の「Amazon CloudWatch Observability EKS アドオンのインストール」を参照してください。 Amazon CloudWatch

CloudWatch エージェントと EKS アドオンに必要な最小バージョンは次のとおりです。

- CloudWatch エージェントバージョン: v1.300045.0
- CloudWatch Observability EKS アドオンバージョン: v2.0.1-eksbuild.1

Amazon CloudWatch コストの見積もり

分割コスト配分データで Amazon CloudWatch Container Insights を使用する機能を有効にすると、Amazon CloudWatch Container Insights に `pod_cpu_usage_total`と の 2 つの新しいメトリクスが追加されます`pod_memory_working_set`。これらのメトリクスの詳細については、「Amazon CloudWatch ユーザーガイド」の「Amazon EKS および Kubernetes Container Insights メトリクス」を参照してください。 Amazon CloudWatch

機能に関連するコストを理解するには

1. https://aws.amazon.com/cloudwatch/pricing/ で Amazon CloudWatch 料金表を開きます。
2. 有料利用枠セクションに移動します。
3. Container Insights タブを選択します。
4. コストの詳細な計算については、「Pricing examples」セクションに移動し、「Example 13 - Container Insights for Amazon EKS and Kubernetes」を参照してください。

レガシー請求レポートについて

このセクションでは、 および AWS Data Exports AWS コストと使用状況レポートの外部で提供されるレガシー請求レポートについて説明します。これらのページは参照用に用意されています。ただし、これらのレポート方法は後日使用できなくなるため、 AWS Data Exports を使用することをお勧めします。

トピック

- 請求明細レポートを使用する
- 請求明細レポートからコストと使用状況レポートへの移行
- 未使用予約のコストの確認
- 月別レポートをダウンロードする
- 毎月のコスト配分レポートをダウンロードする
- AWS 使用状況レポートのダウンロード

請求明細レポートを使用する

> **Important**
>
> 請求明細レポート機能は、2019 年 7 月 8 日現在、新規のお客様にはご利用いただけません。

請求明細レポート (DBR) には、料金に関する AWS コストと使用状況レポート (AWS CUR) と同様の情報が含まれていますが、個々の明細項目の計算方法は異なります。DBR と CUR AWS の両方にサインアップしている場合、明細項目は一致しません。ただし、レポートが月末に確定されると、合計コストは一致します。

AWS は、次の命名規則を使用して DBR を CSV ファイルとして Amazon S3 に保存します。

```
AWS account number-aws-billing-detailed-line-items-yyyy-mm.csv.zip
```

AWS は、詳細請求レポート (DBR) を 1 日に複数回再作成し、レポートを上書きします。がレポートを AWS 上書きすると、明細項目は以前のレポートとは異なる順序になる可能性があります。最終レポートは月末に作成されます。翌月、 は前月の最終レポートを上書きせずに新しいレポートファイル AWS を作成します。過去の月のレポートは、レポートを削除するまで S3 バケットに保存されます。

DBR を CUR AWS に移行する方法については、「」を参照してくださいthe section called “DBR から AWS CUR への移行”。

請求明細レポートからコストと使用状況レポートへの移行

請求明細レポート (DBR) と AWS コストと使用状況レポート (AWS CUR) はどちらも料金に関する情報を提供します。ただし、DBR を使用している場合は、レポートをコストと使用状況レポートに転送することをお勧めします。

トピック

- コストと使用状況レポート (AWS CUR) の利点の比較
- 請求明細レポートとコストと使用状況レポートの主な違い
- アドバンスト料金タイプのレポート

コストと使用状況レポート (AWS CUR) の利点の比較

AWS CUR は、最も包括的な情報源を提供します。 AWS CUR を使用すると、個々のコストを詳細に把握し、より詳細に分析できます。これは、エンタープライズ規模で特に役立ちます。 AWS CUR は、複雑なコスト管理のニーズがあり、専用のクエリまたは分析ベースのシステムが必要な場合に役立ちます。 AWS CUR は、償却コストを含むリザーブドインスタンス (RI) に関する詳細情報も提供します。

包括予約情報

リザーブドインスタンス (RI) または予約は、1 年間または 3 年間のサービス利用契約と引き換えに、オンデマンド利用と比較して割引となる時間料金を提供します。これによりコストを大幅な削減できます。 AWS CUR を使用して予約ポートフォリオをモニタリングおよび管理できます。 AWS CUR は、予約 Amazon リソースナンバー (ARNs)、予約数、合計 RIs などの詳細情報を提供します。特定のリソースに対する予約関連の割引を追跡して、節約額をより深く理解することができます。

詳細請求レポート (DBR) はこのメタデータのサブセットを提供しますが、必要な列を変換する作業が必要です。

AWS CUR は、償却された予約コストに関する情報など、DBR で利用できない追加の列を提供します。詳細については、「the section called “償却された予約データを理解する”」を参照してください。

オンデマンド料金の利用可能性

AWS CUR は、使用量の各明細項目のオンデマンド料金に関する情報を提供します。この情報を使用して、オンデマンド料金から支払った金額を差し引くことで節約額を数値化できます。また、パブリックオンデマンド料金を使用してコストを割り当てるという柔軟な選択もできます。

DBR は、オンデマンド料金の情報を示さず、請求額のみを示します。そのため、全体の節約額の計算や、オンデマンド料金を使用したコストの割り当ては困難です。

割引額の詳細な内訳

AWS CUR は、使用量ベースの割引の詳細なビューにアクセスできます。割引が適用された場合は、CUR AWS を使用して以下を表示できます。

- 割引を受ける前の費用
- 割引額
- 明細項目レベルで割引を適用した後の合計コスト

DBR には、割引の詳細な内訳は含まれていません。

大規模な自動データ取り込み

AWS CUR を使用すると、自動データ取り込みプロセスをトリガーするようにイベントを簡単に設定し、社内システムの請求データを更新するプロセスを合理化できます。前月に関連する料金が検出されたときに、 AWS CUR データを自動的に更新できます。

さらに、 AWS CUR は複数のファイルとして生成されるため、データをより小さな部分にセグメント化する利点があります。こうすることで、複数のワーカーが使用するプロセスに従ってデータを取り込むことができます。さらに、データのダウンロードを少しずつ再試行できます。

AWS CUR は、データをすばやく見つけて抽出できるようにフォーマットされています。このレポートは、データの全体的な構造に関する情報を含むマニフェスト ファイルからモデル化されています。これには、レポートに含まれるすべての列のリストが含まれます。この情報を使用すると、レポートを拡張し、使用状況に関する新しい情報が利用できるようになったときに含めることができます。

製品間統合

AWS CUR は Amazon Redshift、QuickSight、Amazon Athena と統合されています。 AWS CUR を使用して、 AWSベースのコスト管理ソリューションを構築できます。 AWS CUR は Parquet 形式のデータも提供します。これにより、コストと使用状況に関する独自のレポートシステムを構築するた

めのオプションが増えます。詳細については、AWS Billing ユーザーガイドのAWS コストと使用状況レポートのマニフェストファイルを参照してください。

請求明細レポートとコストと使用状況レポートの主な違い

CUR AWS に移行した後に考慮すべき DBR と AWS CUR にはいくつかの違いがあります。たとえば、システムにデータを取り込む方法を調整する必要がある場合があります。

ファイル構造

詳細請求レポート (DBR) は 1 つのファイルとして配信されます。対照的に、 AWS CUR は統合されたファイルセットです。 AWS CUR では、Amazon S3 バケットで次のファイルを表示できます。

- 使用状況行項目をすべて含むデータファイルのセット
- すべての割引を含む別のデータ ファイル (該当する場合)
- 単一のレポートに属するすべてのデータファイルをリストしたマニフェストファイル

列の構造

DBR には列の固定リストがあり、柔軟性が制限されています。 AWS CUR には固定列構造がないため、必要に応じて自由に列を追加または削除できます。新しい の使用を開始すると AWS のサービス、 AWS CUR は動的にレポートに新しいデータを含め始めることができ、これはお客様の場合に役立ちます。マニフェストファイルは、レポート内に存在するすべての列のマップを提供します。

DBR と CUR AWS の同等の列名

DBR 列名	AWS CUR 列名
InvoiceId	bill/InvoiceId
PayerAccountId	bill/PayerAccountId
LinkedAccountId	lineItem/UsageAccountId
ProductName	product/ProductName
SubscriptionId	reservation/subscriptionid
UsageType	lineItem/UsageType
Operation	lineItem/Operation

DBR 列名	AWS CUR 列名
AvailabilityZone	lineItem/AvailabilityZone
ReservedInstance	サポート外
ItemDescription	lineItem/LineItemDescription
UsageStartDate	lineItem/UsageStartDate
UsageEndDate	lineItem/UsageEndDate
UsageQuantity	lineItem/UsageAmount
BlendedRate	lineItem/BlendedRate
BlendedCost	lineItem/BlendedCost
UnBlendedRate	lineItem/UnblendedRate
UnBlendedCost	lineItem/UnblendedCost
ResourceId	lineItem/ResourceId
RecordType	サポート外
Pricingplanid	サポート外
RateID	pricing/RateId

> **Note**
>
> CUR の RecordId AWS に同等のものはありません。 には RecordId に相当する列はありませんが、identity/LineItemId、identity/TimeInterval、および bill/BillType を組み合わせて同等の情報を収集できます。

CUR による DBR RecordType AWS 値の取得

DBR の RecordType 値	CUR を介して RecordType AWS を取得する構文	ユースケース
LineItem	SELECT SUM(line_item_unblended_cost) FROM [CUR] WHERE line_item_line_item_type = 'Usage'	使用状況行項目は、使用コストを 1 回限りの課金から区別します。
InvoiceTotal	SELECT (bill_invoice_id), sum(line_item_unblended_cost) FROM [CUR] GROUP BY bill_invoice_id	請求合計は、請求書とコストと使用状況レポートでコストを一致させるのに役立ちます。
AccountTotal	SELECT line_item_usage_account_id, sum(line_item_unblended_cost) FROM [CUR] GROUP BY line_item_usage_account_id	アカウント合計は、メンバーアカウントに関連するコストを払い戻しの目的で分離できます。
StatementTotal	SELECT SUM(line_item_unblended_cost) FROM [CUR]	ステートメント合計は、請求期間のコストを理解するために役立ちます。
Discount	SELECT SUM(line_item_unblended_cost) FROM [CUR] WHERE line_item_line_item_type = 'Discount'	割引行項目は、割引に関連した行項目を識別するのに役立ちます。
Rounding	まだサポートされていません。	まだサポートされていません。

アドバンスト料金タイプのレポート

Refunds

AWS CUR: 返金は`lineItem/LineItemDescription = ‘Refund’`文字列をフィルタリングすることで識別されます。

DBR: 返金を確認するには ‘Refund’ サブストリングの ItemDescription 列を解析します。

クレジット

AWS CUR: クレジットは`lineItem/LineItemDescription = ‘Credit’`文字列をフィルタリングすることで識別されます。

DBR: クレジットを確認するには ‘Credit’ サブストリングの ItemDescription 列を解析します。

Taxes

AWS CUR: 税は`lineItem/LineItemDescription = ‘Tax’`文字列をフィルタリングすることで識別されます。

DBR: 税金を確認するには ‘Tax’ サブストリングの ItemDescription 列を解析します。

予約関連の前払いコストの確認

AWS CUR: 予約関連の前払いコストは、`"lineItem/LineItemType" = 'Fee'`文字列をフィルタリングすることで識別されます。

DBR: 予約関連の前払い料金を確認するには `'HeavyUsage'` サブストリングの Usagetype 列と、`'SubscriptionId'` が null かどうかを調べます。

予約関連の月額料金の確認

AWS CUR: 予約関連の月額料金は、`"lineItem/LineItemType" = 'RIfee'`文字列をフィルタリングすることで識別されます。

DBR: 予約関連の月額料金を確認するには `'HeavyUsage'` サブストリングの Usagetype 列を解析します。

リザーブドインスタンスのメリットを受けたインスタンスを確認する

AWS CUR: 予約関連の前払い料金は、`"lineItem/LineItemType" = 'DiscountedUsage'`文字列をフィルタリングすることで識別されます。

DBR: 予約関連の初期費用は、ReservedInstance カラムの `'Y'` サブストリングを確認することでわかります。

未使用予約のコストの確認

AWS コストと使用状況レポート (AWS CUR) を使用して、未使用の RI コストを把握できます。次の 4 つのシナリオはその方法を示しています。

> **Note**
>
> 次の表では、わかりやすくするために CUR AWS と DBR/DBR-RT の列と行が変換されています。最初の列の値はレポートのヘッダーを表します。

シナリオ 1: RI 使用率が 100% である場合

RI 料金の明細項目は、未使用コスト $0、使用時間は 0 です。

DBR/DBR-RT では、RI 料金の明細項目の UsageQuantity フィールドと UnblendedCosts フィールドを参照して、未使用の RI の使用状況とコストを確認できます。RI 料金の明細項目は、ItemDescription フィールドに「購入済み時間」の情報があることで識別できます。表 1 は、DBR レポートと DBR-RT レポートで未使用 RI のコストを管理するために使用する列と情報を示しています。

表 1 - 2019 年 6 月 17 日以前の DBR と DBR-RT で RI 使用率が 100% である場合の未使用 RI のコスト

ProductName	Amazon Elastic Compute Cloud	Amazon Elastic Compute Cloud
UsageType	`HeavyUsage:c3.8xlarge`	`HeavyUsage:c3.8xlarge`
操作	`RunInstances`	`RunInstances`
アベイラビリティーゾーン	`us-east-1a`	`us-east-1a`
リザーブドインスタンス	Y	Y
ItemDescription	`USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge (744 hours purchased, 744 hours used)`	`USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge`
使用量	`0`	`744`

非ブレンドレート	0.1	0.1
非ブレンドコスト	0	74.4

AWS CUR を使用すると、RI 料金の明細項目の「reservation/UnusedQuantity」フィールドと「reservation/UnusedRecurringFee」フィールドを参照することで、未使用の RI の使用状況とコストを把握できます。以下の表 4 は、CUR で未使用の RI AWS コストを管理するために使用される現在の列と情報を示しています。

表 2 – AWS CUR で RI 使用率が 100% である場合の未使用 RI のコスト

lineItem/ProductCode	Amazon EC2	Amazon EC2
UsageType	HeavyUsage:c3.8xlarge	USW2-BoxUsage:c3.8xlarge
lineitem/LineItemType	RI Fee	DiscountedUsage
lineitem/LineItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge	USD 0.00 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge
lineitem/UsageAmount	744	744
lineitem/NormalizedUsageAmount	47,616	47,616
lineitem/UnblendedRate	0.1	0
lineitem/UnblendedCost	74.4	0
reservation/UnusedQuantity	0	
reservation/UnusedRecurringFee	0	
reservation/UnusedAmortizedUpfrontFeeForBillingPeriod	0	

reservation/RecurringFeeForUsage		74.4
reservation/AmortizedUpfrontCostForUsage		5
reservation/EffectiveCost		79.4

DBR/DBR-RT でサポートされている現在の機能と一致することに加えて、 AWS CUR には次の利点があります。

- AWS CUR を使用すると、DiscountedUsage 明細項目の EffectiveCost に関する情報にアクセスできます。これには、定期的な料金と前払い料金の両方が含まれます。DBR では、定期的な料金のみ計算されます。
- AWS CUR では、UsageType フィールドは DiscountedUsage 明細項目に対して変換されませんが、DBR は情報を RI 料金明細項目情報に置き換えます。これは、ユーザーがどの使用量がどの RI AWS によってReservationARN で CUR の明細項目をグループ化できるためです。
- AWS CUR では、RI 料金の明細項目に対して LineItemDescription フィールドは変換されません。DBR は、購入した時間数と使用した時間数を追加します。

シナリオ 2: RI を部分的に使用した場合

RI 料金の明細項目には、未使用コストと使用状況が含まれます。

DBR/DBR-RT では、RI 料金の明細項目の UsageQuantity フィールドと UnblendedCosts フィールドを参照して、未使用 RI の使用状況とコストを確認できます。表 3 は、DBR レポートと DBR-RT レポートで未使用 RI のコストを管理するために使用する列と情報を示しています。

表 3 - 2019 年 6 月 17 日以前に DBR と DBR-RT で RI を部分的に使用した場合の未使用 RI のコスト

ProductName	Amazon Elastic Compute Cloud	Amazon Elastic Compute Cloud
UsageType	`HeavyUsage:c3.8xlarge`	`HeavyUsage:c3.8xlarge`

操作	RunInstances	RunInstances
アベイラビリティーゾーン	us-east-1a	us-east-1a
リザーブドインスタンス	Y	Y
ItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge (744 hours purchased, 644 hours used)	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge
使用量	100	644
非ブレンドレート	0.1	0.1
非ブレンドコスト	10	64.4

AWS CUR を使用すると、RI 料金明細項目の「reservation/UnusedQuantity」フィールドと「reservation/UnusedRecurringFee」フィールドを参照することで、未使用の RI の使用状況とコストを把握できます。表 4 は、CUR で未使用の RI AWS コストを管理するために使用される現在の列と情報を示しています。

表 4 – AWS CUR で RI を部分的に使用した場合の未使用 RI のコスト

lineitem/Productcode	Amazon EC2	Amazon EC2
UsageType	HeavyUsage:c3.8xlarge	USW2-BoxUsage:c3.8xlarge
lineitem/LineItemType	RI Fee	DiscountedUsage
lineitem/LineItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge	USD 0.00 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge

lineitem/UsageAmount	744	644
lineitem/NormalizedUsageAmount	47,616	47,216
lineitem/UnblendedRate	0.1	0
lineitem/UnblendedCost	74.4	0
reservation/UnusedQuantity	100	
reservation/UnusedRecurringFee	0	
reservation/UnusedAmortizedUpfrontFeeForBillingPeriod	10	
reservation/RecurringFeeForUsage		64.4
reservation/AmortizedUpfrontCostForUsage		5
reservation/EffectiveCost		69.4

DBR/DBR-RT でサポートされている現在の機能と一致することに加えて、 AWS CUR には次の利点があります。

- AWS CUR には、RI 料金の明細項目と DBR/DBR-RT の UnusedQuantity を表す個別の列があり、UsageQuantity 列を未使用の時間で過負荷にします。

シナリオ 3: キャパシティー予約

RI 料金の明細項目には [UsageQuantity] フィールドと [UnblendedCost] フィールドの未使用金額がすでに含まれているため、RI でカバーする場合、DBR/DBR-RT のキャパシティー予約に関する使用タイプの明細項目 [UnusedBox] および [UnusedDed] は除外されます。表 5 は、DBR レポートと DBR-RT レポートで未使用 RI のコストを管理するために使用する列と情報を示しています。

表 5 - 2019 年 6 月 17 日以前の DBR と DBR-RT でのキャパシティー予約シナリオにおける未使用 RI のコスト

ProductName	Amazon Elastic Compute Cloud	Amazon Elastic Compute Cloud
UsageType	`HeavyUsage:c3.8xlarge`	`HeavyUsage:c3.8xlarge`
操作	`RunInstances`	`RunInstances`
アベイラビリティーゾーン	`us-east-1a`	`us-east-1a`
リザーブドインスタンス	Y	Y
ItemDescription	`USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge (744 hours purchased, 734 hours used)`	`USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge`
使用量	`10`	`734`
非ブレンドレート	`0.1`	`0.1`
非ブレンドコスト	`1`	`73.4`

AWS CUR は、これらの明細項目を DiscountedUsage として表示します。表 6 は、CUR で未使用の RI AWS コストを管理するために使用される現在の列と情報を示しています。

表 6 – CUR でのキャパシティ予約シナリオの未使用の RI AWS コスト

lineitem/Productcode	Amazon EC2	Amazon EC2	Amazon EC2
UsageType	`HeavyUsage:c3.8xlarge`	`USW2-Reservation:c3.8xlarge`	`USW2-BoxUsage:c3.8xlarge`

lineitem/LineItemType	RI Fee	Usage	DiscountedUsage
lineitem/LineItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge	USD 0.00 per Reservation Linux/UNIX (Amazon VPC), c3:8xlarge Instance Hour	USD 0.00 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge
lineitem/UsageAmount	744	744	744
lineitem/NormalizedUsageAmount	47,616		47,216
lineitem/UnblendedRate	0.1	0	0
lineitem/UnblendedCost	74.4	0	0
reservation/RecurringFeeForUsage			64.4
reservation/AmortizedUpfrontCostForUsage			5
reservation/EffectiveCost			69.4

シナリオ 4: サイズの柔軟な予約

DBR/DBR-RT では、RI 料金の明細項目の [UsageQuantity] および [UnblendedCosts] フィールドを参照して、未使用の RI の使用状況やコストを把握することができます。RI 料金の明細項目は、ItemDescription フィールドに「購入済み時間」の情報があることで識別できます。表 9 は、DBR レポートと DBR-RT レポートで未使用 RI のコストを管理するために使用する列と情報を示しています。

表 7 - 2019 年 6 月 17 日以前の DBR と DBR-RT でのサイズ柔軟な RI シナリオにおける未使用 RI のコスト

ProductName	Amazon Elastic Compute Cloud	Amazon Elastic Compute Cloud
UsageType	HeavyUsage:c3.8xlarge	HeavyUsage:c3.8xlarge
操作	RunInstances	RunInstances
アベイラビリティーゾーン	us-east-1a	us-east-1a
リザーブドインスタンス	Y	Y
ItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge (744 hours purchased, 644 hours used)	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge; UsageType : BoxUsage:c3.large
使用量	100	644
非ブレンドレート	0.1	0.1
非ブレンドコスト	10	64.4

AWS CUR を使用すると、RI 料金明細項目の「reservation/UnusedQuantity」フィールドと「reservation/UnusedRecurringFee」フィールドを参照することで、未使用の RI の使用状況とコストを把握できます。表 8 は、CUR で未使用の RI AWS コストを管理するために使用される現在の列と情報を示しています。

表 8 – AWS CUR でのサイズ柔軟な RI シナリオにおける未使用 RI のコスト

lineitem/Productcode	Amazon EC2	Amazon EC2

UsageType	HeavyUsage:c3.8xlarge	USW2-BoxUsage:c3.8xlarge
lineitem/LineItemType	RI Fee	DiscountedUsage
lineitem/LineItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge	USD 0.00 hourly fee per Linux/UNIX (Amazon VPC), c3:8large
lineitem/UsageAmount	744	644
lineitem/NormalizedUsageAmount	47,616	2,576
lineitem/UnblendedRate	0.1	0
lineitem/UnblendedCost	74.4	0
reservation/UnusedQuantity	100	
reservation/UnusedRecurringFee	70.37	
reservation/UnusedAmortizedUpfrontFeeForBillingPeriod	5.5	
reservation/RecurringFeeForUsage		4.03
reservation/AmortizedUpfrontCostForUsage		0.5
reservation/EffectiveCost		4.53

DBR/DBR-RT でサポートされている現在の機能と一致することに加えて、 AWS CUR には次の利点があります。

- AWS CUR には NormalizedUsageAmount と数量があります。DBR/DBR-RT にはこれを表す列はありません。
- AWS CUR UsageType と Operation は、DiscountedUsage 明細項目に対して変換されません。DBR/DBR-RT では、これらの値は、RI 料金の明細項目に置き換わります。
- AWS CUR LineItemDescription は DiscountedUsage 明細項目に対して変換されません。DBR/DBR-RT では、RI 料金の明細項目の説明に置き換わり、[DiscountedUsage] の明細項目 [Usage Type] が文字列 (例:「USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge; UsageType: BoxUsage:c3.large」) の後に追加されます。

月別レポートをダウンロードする

請求情報とコスト管理コンソールの請求書ページから、推定 AWS 請求額の月次レポートをダウンロードすることができます。

一括請求機能を使用するとします AWS Organizations。このレポートは管理アカウントでのみ利用でき、すべてのメンバー アカウントのアクティビティが含まれます。メンバーアカウントの所有者は、管理アカウントからのみ月別レポートを取得できます。詳細については、AWS Billing ユーザーガイド の 組織の一括請求 を参照してください。

レポートには、アカウントが使用する AWS 製品、使用タイプ、オペレーションの一意の組み合わせごとに明細項目が含まれます。見積もりレポートは 1 日に数回更新されます。明細書期間を選択すると、前月のレポートを取得できます。まず、月別レポートに登録した月のレポートから始めます。サインアップする前からのレポートは利用できません。

毎月のコスト配分レポートをダウンロードする

> **Important**
>
> 毎月のコスト配分レポートは後日利用できなくなります。代わりに AWS コストと使用状況レポートを使用することをお勧めします。

AWS 使用量のビジネスディメンションを記述できる AWS リソースのカスタムコスト配分タグセットを作成できます。これらのタグセットを使用して、 AWS コストを整理および追跡できます。多くの は、特徴量セットでタグ付けを AWS のサービス 公開します。これらのサービスでタグを作成するには、コンソール、API、または AWS コマンドラインインターフェイス (CLI) を使用します。詳細については、AWS Billing ユーザーガイド の [コスト配分タグの使用] をご参照ください。

タグを作成すると、毎月のコスト配分レポートを取得できます。これは、基本的にコスト配分タグセットが含まれた月別レポートです。

AWS 使用状況レポートのダウンロード

> **Important**
>
> 2023 年 9 月 15 日、 AWS 使用状況レポートは 2019 年 3 月 1 日より前の使用状況データへのアクセスを提供しなくなります。このような使用状況データにアクセスするには、2023 年 9 月 15 日までに使用履歴をダウンロードしてローカルに保存してください。 AWS 使用状況レポート機能は、後日使用できなくなります。代わりに AWS コストと使用状況レポートを使用することをお勧めします。

使用状況レポートは XMLまたはCSV 形式でダウンロードできます。レポートは、使用タイプ、オペレーション、期間に基づいて、単一のサービスを対象としています。また、データの収集方法も選択できます。

使用状況レポートのダウンロード

1. https://console.aws.amazon.com/costmanagement/ で Billing and Cost Management コンソールを開きます。
2. ナビゲーションペインのレガシーページで、コストと使用状況レポートを選択します。
3. [AWS 使用状況レポート]セクションで、[使用状況レポートの作成] を選択します。
4. 「使用状況レポートのダウンロード」ページの「サービス」で、使用状況を確認したいサービスを選択します。
5. 使用タイプを選択します。
6. [操作]を選択します。
7. レポートの期間を選択します。カスタム日付範囲を選択した場合は、レポートの日付範囲を手動で指定する必要があります。
8. [レポートの精度] で、[時間単位]、[日単位]、または [月単位] を選択します。
9. [ダウンロード] を選択し、[XML レポート] または [CSV レポート] を選択します。

> **Note**
>
> サイズの大きいレポートをダウンロードすると、レポートの内容が切り捨てられることがあります。ダウンロードしたファイルの最後の行に警告やエラーメッセージがないか確認してください。レポートが切り捨てられている場合は、短い期間を選択して小さいレポートをダウンロードしてください。もう 1 つの選択肢は、レポートの粒度を時間単位から日単位または月単位に減らすことです。

コストと使用状況レポートのトラブルシューティング

以下のトピックを使用して、コストと使用状況レポートの一般的な問題のトラブルシューティングを支援します。

トピック

- Amazon S3 バケットにはレポートファイルはありません
- レポートデータパーティションの 1 つが空です
- コストと使用状況レポートのデータが、他の請求とコスト管理機能のデータと一致しない
- レポートの設定を変更したので、データをバックフィルしたいです。
- Amazon S3 のレポートファイルフォルダが名前のないフォルダにあります
- レポートにリソース ID を含めるオプションを選択できない
- Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で動作しない、または Amazon Redshift クエリが Amazon Athena で動作しない
- レポートに含まれる列が前月と変わっています。
- レポート内の列が変更されたため、レポートに基づくクエリやテーブルが機能しない
- レポートをクエリする方法を知りたい
- Amazon EC2 専用ホストの請求データが見つかりません
- Amazon EC2 Elastic IP アドレスの請求データがわかりません
- 一括請求を使っているが、非ブレンドとブレンドのレートまたはコストの違いがわからない
- レポートの一部の明細項目には、ブレンドレートまたはブレンドコストが 0 のものがあります。
- 私のレポートでは、全額前払いリザーブドインスタンスがどのように償却されているのかわかりません。

Amazon S3 バケットにはレポートファイルはありません

Amazon S3 バケットポリシーが、billingreports.amazonaws.com サービスにバケットにファイルを配置するアクセス許可を付与していることを確認します。必要なバケットポリシーの詳細については、「コストと使用状況レポート用に Amazon S3 バケットをセットアップする」を参照してください。

レポートデータパーティションの 1 つが空です

レポートがほとんどのアプリケーションが処理できるサイズより大きい場合、 はレポートを複数のファイルに AWS 分割します。レポートを更新すると、個々のファイルパーティションが以前のレポートバージョンよりも少なくなる可能性があります。

レポートのマニフェストファイルを確認して、取り込む必要のない空のファイルを見つけてください。

コストと使用状況レポートのデータが、他の請求とコスト管理機能のデータと一致しない

その他の請求とコスト管理機能 (Cost Explorer、請求詳細レポート、請求とコスト管理コンソール) では、以下の理由によりコストの表示が異なる場合があります。

- 請求機能では、コストデータをさまざまな方法で四捨五入します。
- 請求機能ではデータ更新設定が異なる場合があります。たとえば、コストと使用状況レポートで、以前に決済された請求書を、請求書の確定後に適用された返金、クレジット、またはSupport 料金で自動的に更新するかどうかを選択できます。Cost Explorer には同じ項目が自動的に反映されます。このシナリオでは、コストと使用状況レポートの自動更新を有効にしないと、コストと使用状況レポートのデータは Cost Explorer のデータと一致しません。
- 請求機能では料金を異なる方法でグループ化できます。たとえば、Billing and Cost Management コンソールの請求書ページには、データ転送料金が AWS サービス料金 内の個別の データ転送 グループとして表示されます。一方、コストと使用状況レポートとCost Explorer には、各サービスの使用タイプとしてデータ転送料金が表示されます。

これらの理由を確認しても、コストと使用状況レポートとその他の請求およびコスト管理機能との間に相違があると思われる場合は、サポートケースを開いてコストデータの確認をリクエストしてください。サポートケースでは、確認するレポート名と請求期間を必ず提供してください。 ケースを開く方法については、「エクスポートとレポートに関するヘルプの取得」を参照してください。

レポートの設定を変更したので、データをバックフィルしたいです。

サポートケースを開いて、コストデータのバックフィルをリクエストします。サポートケースでは、バックフィルするレポート名と請求期間を必ず指定してください。 ケースを開く方法については、「エクスポートとレポートに関するヘルプの取得」を参照してください。

以下のシナリオでは、コストデータのバックフィルを取得できないことに注意してください。

- アカウントを作成した日より前のコストデータのバックフィルは取得できません。
- を使用し AWS Organizations 、管理アカウントとして指定されているアカウントなど、組織の構造が変更された場合、以前の組織構造でデータのバックフィルを取得することはできません。
- を使用して組織 AWS Organizations を変更した場合、現在の組織に参加する前に からデータのバックフィルを取得することはできません。

Amazon S3 のレポートファイルフォルダが名前のないフォルダにあります

レポートのレポートパスプレフィックス に / 文字があると、Amazon S3 バケットに名前のないフォルダが生成されます。次回のレポート更新時に名前のないフォルダを削除するには、レポート設定を編集し、レポートパスのプレフィックスから / 文字を削除します。手順については、「コストと使用状況レポートの設定の編集」を参照してください。

レポートにリソース ID を含めるオプションを選択できない

レポートを作成するときに、リソース ID を含める オプションを選択できます。[レポートバージョニング] を [既存のレポートを上書き] に設定してレポートを作成した場合、レポートを作成した後に [リソース ID を含める] の選択を変更することはできません。リソース ID を含めるには、新しいレポートを作成して [リソース ID を含める] オプションを選択する必要があります。

Amazon Athena のコストと使用状況レポートのクエリが Amazon Redshift で動作しない、またはAmazon Redshift クエリが Amazon Athena で動作しない

Amazon Athena データベースと Amazon Redshift データベースでは、コストと使用状況レポートの列の形式が異なります。Amazon Athena では、列名 (line_item_normalized_usage_amount) の単語の間にアンダースコアが追加されます。Amazon Redshift では、列タイプと属性 (lineitem_normalizedusageamount) の間にアンダースコアが追加されます。Amazon Athena または Amazon Redshift の列名の形式と一致するようにクエリを変更してください。

レポートに含まれる列が前月と変わっています。

がレポート AWS に含める列は、 AWS 使用状況によって異なります。すべてのレポートには、identity/、bill/、lineItem/ というプレフィックスが付いている列が含まれています。

- identity/LineItemId
- identity/TimeInterval
- bill/InvoiceId
- bill/BillingEntity
- bill/BillType
- bill/PayerAccountId
- bill/BillingPeriodStartDate
- bill/BillingPeriodEndDate
- lineItem/UsageAccountId
- lineItem/LineItemType
- lineItem/UsageStartDate
- lineItem/UsageEndDate
- lineItem/ProductCode
- lineItem/UsageType
- lineItem/Operation
- lineItem/AvailabilityZone
- lineItem/ResourceId
- lineItem/UsageAmount
- lineItem/NormalizationFactor
- lineItem/NormalizedUsageAmount
- lineItem/CurrencyCode
- lineItem/UnblendedRate
- lineItem/UnblendedCost
- lineItem/BlendedRate
- lineItem/BlendedCost
- lineItem/LineItemDescription
- lineItem/TaxType

- lineItem/LegalEntity

他のすべての列は、毎月の AWS 使用量がそれらの列に入力するデータを生成する場合にのみ含まれます。

例えば、その月に Savings Plans を使用した場合のみ、レポートに savingsPlan/ 列が含まれます。

レポート内の列が変更されたため、レポートに基づくクエリやテーブルが機能しない

がレポート AWS に含める列は、その月の AWS 使用状況によって異なります。レポートに含まれる列は変更される可能性があるため、レポートに基づくカスタムクエリやテーブルでは、列番号の代わりに列名を参照するのがベストプラクティスです。

レポートをクエリする方法を知りたい

コストと使用状況レポートのクエリの詳細については、 AWS Well-Architected Labs ウェブサイトの「CUR クエリライブラリのヘルプ」を参照してください。

Amazon EC2 専用ホストの請求データが見つかりません

ResourceID 列で、インスタンス ID ではなく専用ホスト ID を探します。Dedicated Hosts は Dedicated Host の実行時間によって計測されるため、レポートには Dedicated Host の使用状況が Dedicated Host の使用状況を、ホスト ID に関連付けられた従量制時間別に表示します。

Amazon EC2 Elastic IP アドレスの請求データがわかりません

Amazon EC2 Elastic IP アドレスは集計して計測されます。つまり、レポートの各行項目が個々の Elastic IP アドレスに対応しているわけではありません。各行項目は課金対象時間の合計を表します。実行中のインスタンスに 1 つの Elastic IP アドレスを無料で割り当てることができます。インスタンスに割り当てる Elastic IP アドレスが追加されるたびに、1 時間ごとに比例配分して課金されます。さらに、 は割り当てられていない Elastic IP アドレスに対して時間単位の料金を AWS 請求します。

一括請求を使っているが、非ブレンドとブレンドのレートまたはコストの違いがわからない

の一括請求では AWS Organizations、非ブレンドレートとブレンドレートまたはコストが、組織内のスタンドアロンアカウントとリンクされたアカウントのアカウントの使用コストを理解するのに役立

ちます。一部のサービスでは、使用量が増えるにつれて単価を下げることができる価格帯が提供されています。は組織内のサービスのすべての使用量を AWS 集計するため、個々のアカウントは、使用量が組織の毎月の使用量に集計されると、より早く低価格の階層にアクセスする可能性があります。

非ブレンドレートは、個々のアカウントのサービス使用量に関連する料金です。詳細項目の場合、非ブレンドコストは使用量に非ブレンドレートを掛けたものです。スタンドアロンアカウントの場合、非ブレンドコストはアカウントの使用コストになります。ブレンドレートは、組織内の総使用量に関連する料金をアカウント全体で平均したものです。詳細項目の場合、ブレンドコストは使用量にブレンドレートを掛けたものです。ブレンドコストは、組織内の連結アカウントとしてのアカウントの使用状況に起因するコストです。

非ブレンド費用とブレンド費用の計算の詳細と例については、AWS Billing ユーザーガイド の「一括請求について」を参照してください。

レポートの一部の明細項目には、ブレンドレートまたはブレンドコストが 0 のものがあります。

リザーブドインスタンスdiscount のある Amazon EC2 品目のブレンドレートはゼロです。これらの詳細項目の場合、LineItemType は「割引使用」です。

ブレンドコストは、使用量にブレンドレートを掛けたものです。ブレンドレートまたは使用量の値が 0 の場合、ブレンドコストも 0 になります。

私のレポートでは、全額前払いリザーブドインスタンスがどのように償却されているのかわかりません。

全額前払いリザーブドインスタンスは全額前払いで支払われるため、償却された費用は、関連する期間 (1 年または 3 年) にわたって分割された前払いとしてレポートに反映されます。

reservation/AmortizedUpfrontCostForUsage および reservation/EffectiveCost は、全額前払いリザーブドインスタンスで同じレートです。これは、どちらの列も、リザーブドインスタンスの前払い金をその期間の合計時間に対して等しく割ったものだからです。

RiFee が 0.00 USD であっても、レポートには全額前払いリザーブドインスタンスの RiFee 明細項目が入力されていることが予想されます。これらの項目は、その月の定期的な時間単位のコストを表し、他の列には追加の使用量データがあります。すべてのリザーブドインスタンスは RiFee の明細項目を生成します。

AWS コストと使用状況レポートのセキュリティ

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。

AWS コストと使用状況レポートは、 AWS Billing and Cost Management コンソールの機能です。セキュリティに関する考慮事項の詳細については、AWS Billing ユーザーガイドの AWS Billing and Cost Managementのセキュリティを参照してください。

CUR を使用するためのアクセスコントロールと IAM AWS アクセス許可の詳細については、「アクセス許可の管理の概要」を参照してください。

クォータと制限

次の表は、 AWS コストと使用状況レポート内の現在のクォータと制限を示しています。

コストと使用状況レポート

コストと使用状況レポートの数	アカウントあたり 10 個
レート	コストと使用状況レポートは無料ですが、標準の Amazon S3 料金が適用されます。
無料利用枠の AWS コストと使用状況レポートの数	10

エクスポートとレポートに関するヘルプの取得

AWS Billing and Cost Management、、および AWS コストと使用状況レポートに関する質問に関するヘルプは AWS Data Exports、多くのリソースから入手できます。

- AWS ナレッジセンター: これは、エクスポートとレポートに関する質問に対する回答を見つける最も簡単な方法です。ここから始めることをお勧めします。
- アカウントと請求のサポート: AWS アカウント所有者は、アカウントと請求のサポートに無料でアクセスできます。個別の技術サポートのみがサポートプランを必要とします。詳細については、「サポート」を参照してください。
- サポートケースを開く: に連絡して、問い合わせのサポートケース AWS サポート を開くことができます。これは、 と通信するための最も直接的な方法です AWS サポート。 は、担当者に連絡するための直通電話番号を発行 サポート せず、代わりに次の手順で電話します。

> **Note**
>
> サポート ケースを開き、「アカウントと請求サポートについて」と指定するには、ルートアカウントの所有者 AWS として にサインインするか、サポートケースを開くための IAM アクセス許可が必要です。詳細については、「サポート ユーザーガイド」の「サポートの使用開始」を参照してください。
> AWS アカウントを閉鎖した場合でも、 にサインイン サポート して過去の請求書を表示できます。

に問い合わせるには AWS サポート

1. サインインしてサポート センターに移動します。
2. [ケースを作成] を選択します。
3. [ケースを生成] ページで、[アカウントと請求] を選択し、フォームの必須フィールドに入力します。
4. [Next step: Additional information] (次のステップ:追加情報) を選択します。
5. 追加情報ページの件名に、問題に関するタイトルを入力します。
6. 説明 に、質問または問題の詳細を入力します。

7. (オプション) 添付ファイルを選択して、エラーログやスクリーンショットなど、関連するファイルをケースに追加します。最大 3 つまでのファイルをアタッチできます。各ファイルは、最大 5MB まで可能です。
8. 次のステップ:今すぐ解決するか、お問い合わせくださいを選択します。
9. [Contact us] (お問い合わせ) ページで、希望する言語を選択します。
10. 希望する連絡方法を変更します。次のオプションのいずれかを選択します。

    - ウェブ サポートセンターで返信を受け取ります。
    - 電話: サポート 担当者から電話してもらいます。

    > **Note**
    >
    > インスタントメッセージは、請求に関するお問い合わせにはご利用いただけません。

11. ケースの詳細を確認して、[Submit] (送信) を選択します。ケース ID 番号と概要が表示されます。

ドキュメント履歴

次の表に、 のこのリリースのドキュメントを示します AWS Data Exports。

変更	説明	日付
炭素排出量データのエクスポートを追加	データエクスポートで炭素排出量データのエクスポートを作成できます。	2025 年 4 月 23 日
GA で FinOps 基盤オープンソース (列を含む AWS FOCUS 1.0) を使用したエクスポートを追加	一般提供 (GA) の FinOps Open Cost and Usage Specification (FOCUS) を使用して、データエクスポートでコストと使用状況のエクスポートを作成できます。	2024 年 11 月 25 日
分割コスト配分データと Amazon CloudWatch Container Insights	Amazon EKS の分割コスト配分データは、Amazon CloudWatch Container Insights のメトリクスをサポートするようになりました。	2024 年 11 月 14 日
FinOps 基盤のオープンソース (AWS 列を含む FOCUS 1.0 - プレビュー) を使用したエクスポートの追加	FinOps Open Cost and Usage Specification (FOCUS) を使用して、データエクスポートでコストと使用状況のエクスポートを作成できます。	2024 年 6 月 20 日
コスト最適化の推奨事項のエクスポートの追加 (Cost Optimization Hub から)	データエクスポートでコスト最適化の推奨事項のエクスポート (Cost Optimization Hub から) を作成できます。	2024 年 6 月 20 日
分割コスト配分データと Amazon Managed Service for Prometheus	Amazon EKS の分割コスト配分データは、Amazon Managed Service for Prometheus のメトリクスを	2024 年 6 月 10 日

	サポートするようになりました。	
Amazon EKS との分割コスト配分データ統合	分割コスト配分データ機能を Amazon EKS と統合するように拡張しました。	2024 年 4 月 16 日
AWS Data Exports 起動	AWS Data Exports では、列の選択と行フィルタリングに SQL を使用して、コストと使用状況レポート (CUR) 2.0 のエクスポートを作成できます。これは、 AWSから最も詳細なコストと使用状況の詳細を取得するための新しい推奨方法です。	2023 年 11 月 26 日

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。