



Add a permission의

AWS Data Exports



AWS Data Exports: Add a permission의

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

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

AWS Data Exports란 무엇인가요?	1
데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션	3
CUR 스키마를 사용하여 내보내기 생성	5
새 스키마를 사용하여 CUR 2.0의 내보내기 생성	7
데이터 내보내기 생성	9
Amazon S3 버킷에 데이터 내보내기 설정	10
표준 내보내기 생성	12
Cost and Usage Dashboard 생성	15
레거시 CUR 내보내기 생성	17
결제 보기를 사용하여 내보내기 생성	19
데이터 쿼리-SQL 쿼리 및 테이블 구성	20
SQL 쿼리	21
테이블 구성	23
Billing Conductor를 사용하여 AWS CUR 2.0 구성	24
표준 및 AWS Billing Conductor 비용 및 사용 보고서 비교	24
결제 그룹에 대한 견적 비용 및 사용 보고서 생성	24
데이터 내보내기 보기 및 관리	26
내보내기 전송에 대한 이해	27
내보내기 S3 상위 디렉터리 구조	27
내보내기 새로 고침	28
내보내기 덮어쓰기 및 새로 생성	29
내보내기 데이터 파일 이름 및 청크	30
요약	30
내보내기 세부 정보 편집	31
내보내기 태그 편집	32
내보내기 삭제	32
AWS 조직에서 데이터 내보내기 사용	33
데이터 내보내기 테이블 사전	35
비용 및 사용량 보고서(CUR) 2.0	35
테이블 구성	36
AWS 조직 지원	39
CUR 2.0 열 그룹	40
청구서 열	40
비용 범주 열	42

할인 열	42
ID 열	46
품목 열	47
요금 열	57
제품 열	59
예약 열	63
리소스 태그 열	70
절감형 플랜 열	70
분할 품목 열	73
태그 열	79
용량 예약 열	81
비용 최적화 권장 사항	85
테이블 구성	85
서비스 연결 역할	86
AWS 조직 지원	87
비용 최적화 권장 사항 열	87
AWS 열이 있는 FOCUS 1.2	90
테이블 구성	90
AWS 조직 지원	90
AWS 열이 있는 FOCUS 1.2	91
AWS 열 적합성 격차가 있는 FOCUS 1.2	100
FOCUS 1.0에서 FOCUS 1.2로 마이그레이션	103
AWS 열이 있는 FOCUS 1.0	105
테이블 구성	105
AWS 조직 지원	106
AWS 열이 있는 FOCUS 1.0	106
AWS 열 적합성 격차가 있는 FOCUS 1.0	113
Cost and Usage Dashboard	115
테이블 구성	115
AWS 조직 지원	115
Cost and Usage Dashboard 열	115
탄소 배출량	127
기록 데이터	127
테이블 구성	127
권한	127
모델 버전	127

AWS 조직 지원	127
탄소 배출량 열	128
데이터 내보내기 처리	133
Amazon Athena 구성	133
Amazon Redshift 구성	134
CUR 2.0 처리를 위한 권장 SQL 쿼리	134
탄소 배출량 데이터 내보내기 처리를 위한 권장 SQL 쿼리	134
Cost and Usage Dashboard의 이해	136
비용 및 사용량 보고서(CUR)의 이해	137
탄소 배출량 데이터 내보내기 이해	138
보안 및 권한	139
데이터 내보내기에 대한 ID 및 액세스 관리	139
샘플 정책	143
데이터 내보내기의 데이터 보호	145
S3 보안 모범 사례	145
S3에서 데이터 암호화	145
할당량 및 제한	146
문제 해결	147
일반 문제 해결	147
내보내기 상태가 비정상인 이유는 무엇인가요?	147
Data Exports에서 내 SQL 문을 수락하지 않는 이유는 무엇인가요?	149
데이터 내보내기 내에서 Athena를 구성하기 위해 미리 정의된 SQL 스크립트를 찾을 수 없습 니다. 그 이유는 무엇인가요?	149
내보내기 파티션 중 하나가 비어 있는 이유는 무엇인가요?	149
Amazon S3 버킷에 보고서 파일이 없는 이유는 무엇인가요?	149
CUR 2.0 문제 해결	149
CUR에서 사용 가능했던 일부 열이 CUR 2.0에서는 표시되지 않는 이유는 무엇인가요?	150
레거시 비용 및 사용 보고서는 어떻게 되나요? 사용이 중단되나요?	150
CUR 2.0 내보내기를 생성하면 레거시 CUR에 영향을 미치나요?	151
데이터 내보내기 및 CUR 테이블을 사용할 수 있는 IAM 권한이 있음에도 CUR 2.0 내보내 기를 생성할 수 없는 이유는 무엇인가요?	151
레거시 CUR 열과 동일한 CSV 형식으로 데이터 내보내기를 생성하려고 하면 "Invalid QueryStatement" 오류가 발생합니다. 이 문제를 해결하려면 어떻게 해야 하나요?	151
데이터 내보내기 CUR 2.0으로 마이그레이션한 후 레거시 CUR 내보내기와 CUR 2.0 내보내 기를 동시에 사용할 수 있나요?	151

CUR 2.0의 내보내기를 생성하려고 할 때 "이 계정은 이 테이블에 대한 내보내기를 생성할 수 없습니다"라는 오류가 발생합니다. CUR 2.0 내보내기를 생성할 수 없는 이유는 무엇인가요?	151
Cost and Usage Dashboard 문제 해결	151
Cost and Usage Dashboard를 만든 직후에 내보내기가 실패하는 이유는 무엇인가요?	152
대시보드에 액세스할 수 없는 이유는 무엇인가요?	152
대시보드를 보려고 할 때 QuickSight 계정 구독을 취소하라는 콘솔 관리 페이지로 이동하는 이유는 무엇인가요?	152
방금 만든 Cost and Usage Dashboard에 이번 달의 데이터가 보이지 않는 이유는 무엇인가요?	152
Cost and Usage Dashboard에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?	153
QuickSight 대시보드 링크가 Data Exports 콘솔 페이지에서 사라진 이유는 무엇인가요?	153
CUR 2.0에서 리소스 태그를 시각화하도록 Amazon QuickSight를 구성하려면 어떻게 해야 하나요?	153
비용 및 사용 보고서 문제 해결	154
비용 및 사용 보고서 데이터가 다른 Billing and Cost Management 특성에서 표시된 데이터와 일치하지 않는 이유는 무엇인가요?	154
보고서 설정을 변경한 후 데이터를 어떻게 백업할 수 있나요?	155
Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있는 이유는 무엇인가요? ..	155
보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없는 이유는 무엇인가요?	155
Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않는 이유는 무엇인가요?	156
보고서에 포함된 열이 전월과 달라진 이유	156
보고서의 열이 변경된 후 쿼리 또는 테이블이 작동하지 않는 이유는 무엇인가요?	157
보고서를 쿼리하려면 어떻게 해야 하나요?	157
Amazon EC2 전용 호스트의 결제 데이터는 어디에서 찾을 수 있나요?	157
Amazon EC2 탄력적 IP 주소의 결제 데이터는 어떻게 해석해야 하나요?	157
통합 결제에서 일반 요금과 혼합된 요금 또는 비용은 어떻게 다를까요?	158
보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0 이유는 무엇인가요?	158
보고서에서 전체 선결제 예약 인스턴스는 어떻게 분할 상환되나요?	158
탄소 배출량 데이터 내보내기 문제 해결	159
IAM 권한으로 데이터 내보내기 및 CUR 2.0 테이블 사용이 가능함에도 불구하고, 탄소 배출량 테이블의 내보내기를 생성할 수 없는 이유는 무엇인가요?	159
조직 내 일부 멤버 계정의 탄소 배출량 데이터를 볼 수 없는 이유는 무엇인가요?	159
S3 버킷의 파일 중 하나가 비어 있는 이유는 무엇인가요?	160

사용 데이터가 있는 경우 일부 리전 및 서비스에서 S3 내보내기에 탄소 배출량이 0으로 표시되는 이유는 무엇인가요?	160
데이터 내보내기에서 과거 데이터 채우기를 탄소 배출량에 사용할 수 있나요?	160
보고서 설정을 변경한 후 또는 새 방법론이 릴리스될 때 데이터를 채우려면 어떻게 해야 하나요?	160
S3 버킷에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?	160
내보내기에서 새로 추가된 열이 보이지 않는 이유는 무엇인가요?	161
이전 방법론 버전을 사용하여 내 데이터가 제공되지 않는 이유는 무엇인가요?	161
레거시 비용 및 사용량 보고서	162
AWS 비용 및 사용 보고서란 무엇입니까?	162
비용 및 사용 보고서 작동 방식	162
보고서 타임라인	163
보고서 파일	163
보고서 열	164
보고서 사용	164
비용 및 사용 보고서 생성	164
비용 및 사용 보고서에 대한 Amazon S3 버킷 설정	165
보고서 생성	167
보고서 보기 및 관리	169
최신 보고서 버전 보기	169
최종 보고서 보기	170
보고서 버전 이해	171
보고서 편집	175
에 대한 비용 및 사용 보고서 사용 AWS Organizations	177
Athena를 사용하여 보고서 쿼리	178
CloudFormation으로 Athena 설정	179
Athena 수동 설정	181
Athena 쿼리 실행	184
기타 리소스	185
Billing Conductor를 사용하여 AWS CUR 구성	187
Billing Conductor AWS CUR과 표준 AWS CUR의 차이점	188
결제 그룹을 위한 견적 비용 및 사용 보고서 생성	188
데이터 사전	190
자격 증명 세부 정보	190
결제 세부 정보	191
항목 세부 정보	193

예약 세부 정보	199
요금 세부 정보	209
제품 세부 정보	211
리소스 태그 세부 정보	242
절감형 플랜 세부 정보	242
비용 범주 세부 정보	248
할인 세부 정보	249
분할 항목 세부 정보	249
사용 사례	254
절감형 플랜 이해	254
예약 파약	258
데이터 전송 요금 이해	270
분할 비용 할당 데이터에 대한 이해	273
레거시 결제 보고서 이해	300
세부 결제 보고서	300
DBR에서 AWS CUR로 마이그레이션	301
미사용 예약 비용 이해	307
월별 보고서	316
월별 비용 할당 보고서	316
AWS 사용 보고서	317
문제 해결	318
Amazon S3 버킷에 보고서 파일이 없음	318
보고서 데이터 파티션 중 하나가 비어 있음	318
내 비용 및 사용 보고서 데이터가 다른 결제 및 비용 관리 기능의 데이터와 일치하지 않음	319
보고서 설정을 변경했기 때문에 데이터를 채우고 싶음	319
Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있음	320
보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없음	320
Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않음	320
보고서에 포함된 열이 전월과 달라짐	320
보고서의 열이 변경되어 보고서에 기반한 쿼리나 테이블이 작동하지 않음	321
보고서를 쿼리하는 데 도움이 필요함	321
Amazon EC2 전용 호스트의 결제 데이터를 찾을 수 없음	322
Amazon EC2 탄력적 IP 주소의 결제 데이터를 이해할 수 없음	322
통합 결제를 사용하는데 일반 요금 및 혼합 요금이나 비용의 차이를 이해할 수 없음	322
보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0임	322

보고서에서 전체 선결제 예약형 인스턴스가 분할 상환되는 방식을 이해할 수 없음	323
보안 및 권한	323
할당량 및 제한	323
비용 및 사용 보고서	323
도움말 가져오기	324
문서 이력	326
.....	CCCXXIX

AWS Data Exports란 무엇인가요?

AWS Data Exports 를 사용하면 기본 SQL을 사용하여 결제 및 비용 관리 데이터 내보내기 및 탄소 배출량 데이터 내보내기를 생성하고 Amazon QuickSight와 통합하여 데이터를 시각화할 수 있습니다.

AWS 결제 및 비용 관리 콘솔, AWS CLI 또는 AWS SDK를 사용하여 내보내기를 생성할 수 있습니다. 콘솔에서 사용자 지정 열 선택을 사용할 수 있습니다. AWS CLI 또는 AWS SDK에서 SQL 쿼리를 작성하고, 열을 선택하고, 행을 필터링하고, 열의 이름을 바꿀 수 있습니다. 이를 통해 처리하려는 데이터만 선택하고, 민감한 비용 정보를 제거하고, 내보내기의 출력 스키마를 제어할 수 있습니다.

다음과 같은 5가지 내보내기 유형이 있습니다.

- 표준 데이터 내보내기, 4개의 테이블 중에서 선택할 수 있습니다.
- 비용 및 사용량 보고서 2.0(CUR 2.0)

Note

비용 및 사용 보고서 2.0(CUR 2.0)은 상세한 AWS 비용 및 사용 데이터를 수신하는 새롭고 권장되는 방법입니다. CUR 2.0은 이전 비용 및 사용량 보고서(CUR)와 비교할 때 몇 가지 개선 사항이 포함되어 있습니다. 자세한 내용은 [데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션](#)을 참조하세요.

- 비용 최적화 권장 사항(Cost Optimization Hub에서)
- AWS 열이 있는 FOCUS 1.2
- AWS 열이 있는 FOCUS 1.0
- 탄소 배출량
- Cost and Usage Dashboard: Amazon QuickSight로 내보내고 통합하여 사전 구축된 Cost and Usage Dashboard를 배포합니다.
- 레거시 데이터 내보내기: 레거시 비용 및 사용량 보고서(CUR)의 내보내기입니다. 그러나 레거시 CUR은 데이터 내보내기 작업([데이터 내보내기 작업 참조](#) 확인)과는 다른 작업 세트([CUR 작업 참조](#) 확인)로 액세스합니다.

데이터 내보내기에는 다음과 같은 이점이 있습니다.

- 제공되는 가장 세분화된 데이터를 사용하여 반복 내보내기를 생성하여 Amazon S3에 저장합니다.
- 열 선택 및 행 필터를 사용하여 SQL 쿼리로 데이터 내보내기를 사용자 지정합니다.

- 원하는 열만 포함하는 일관된 스키마를 사용하여 내보내기를 생성합니다.
- 연결된 특정 AWS 계정 IDs와 관련된 민감한 비용 데이터 또는 요금을 제거합니다.
- 필요한 열 또는 행만 선택하여 내보내기 크기를 줄입니다.
- 다운스트림 분석을 지원하기 위해 비용 데이터 및 탄소 발자국 데이터의 자동화된 내보내기를 구현합니다.

데이터 내보내기를 시작하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.

데이터 내보내기 페이지에서 새 내보내기를 생성하고, 기존 내보내기를 관리하고, Amazon QuickSight 와 통합되고 사전 구축된 Cost and Usage Dashboard를 배포하는 내보내기를 생성할 수 있습니다.

데이터 내보내기 페이지에서 [AWS 지속 가능성](#) 및 [AWS 사용 보고서에](#) 액세스할 수도 있습니다.

데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션

AWS Data Exports 를 사용하면 비용 및 사용 보고서 2.0(CUR 2.0)의 내보내기를 생성할 수 있습니다. CUR 2.0 테이블은 비용 및 사용량 보고서(CUR)와 동일한 정보와 함께 몇 가지 개선 사항을 제공합니다. 데이터 내보내기를 사용하면 CUR을 처리하는 데 사용했던 데이터 파이프라인과 역호환되는 CUR 2.0 내보내기를 생성할 수 있습니다.

CUR 2.0은 CUR에 비해 다음과 같은 향상된 기능을 제공합니다.

- 일관된 스키마: CUR 2.0에는 고정된 열 세트가 포함되지만 CUR에 포함되는 열은 AWS 서비스 사용량, 비용 범주 및 리소스 태그에 따라 매월 달라질 수 있습니다.
- 중첩 데이터: CUR 2.0은 CUR의 열의 키-값 쌍이 있는 개별 열로 특정 열을 축소하여 데이터 희소성을 줄입니다. 원하는 경우 데이터 내보내기에서 중첩된 키를 별도의 열로 쿼리하여 원래 CUR 스키마 및 데이터와 일치시킬 수 있습니다.
- 추가 열: CUR 2.0에는 `bill_payer_account_name`과 `line_item_usage_account_name`이라는 두 개의 추가 열이 추가되었습니다.

다음 테이블에서는 CUR 2.0과 레거시 CUR의 차이점을 자세히 설명합니다.

	CUR 2.0	레거시 CUR
데이터 스키마	스키마를 수정했습니다. 전체 열 목록은 비용 및 사용량 보고서(CUR) 2.0 을 참조하세요.	AWS 사용량 및 활동을 기반으로 하는 동적 스키마입니다. 부분 열 목록은 데이터 사전 을 참조하세요.
전용 열	<code>bill_payer_account_name</code> <code>line_item_usage_account_name</code>	없음
사용자 지정 내보내기	열 선택, 행 필터링 및 열 별칭 지정(이름 변경)에 기본 SQL을 활성화합니다.	지원되지 않음. 필요한 보기를 생성하려면 Athena/QuickSight를 수동으로 설정해야 합니다.

	CUR 2.0	레거시 CUR
	지원되는 SQL 구문에 대한 자세한 내용은 데이터 쿼리 섹션을 참조하세요.	
키-값 페어 포함 중첩된 열	resource_tags cost_category product discount	중첩된 열이 없습니다. CUR 2.0의 중첩된 열 4개는 레거시 CUR의 개별 열(예: resource_tags_user_creator)로 분할됩니다.
파일 전송 대상	S3 버킷	S3 버킷
파일 출력 형식	GZIP, Parquet	ZIP, GZIP, Parquet
다른 AWS 서비스와의 통합	Amazon QuickSight	Amazon Athena, Amazon Redshift, Amazon QuickSight
Amazon CloudFormation 지원	예 자세한 내용은 AWS CloudFormation 사용 설명서의 AWS 데이터 내보내기 리소스 유형 참조 를 참조하세요.	예 자세한 내용은 AWS CloudFormation 사용 설명서의 AWS 비용 및 사용량 보고서 리소스 유형 참조 를 참조하세요.

	CUR 2.0	레거시 CUR
태그 및 비용 범주 데이터	태그 및 비용 범주 이름은 특수 문자와 공백을 제거하기 위해 정규화됩니다. 정규화 후 충돌하는 태그 또는 비용 범주가 있을 경우 하나의 값만 유지됩니다. 자세한 내용은 열 이름 을 참조하세요.	동작은 레거시 CUR Parquet와 CSV 파일 형식 간에 다릅니다. 레거시 CUR Parquet: 태그 및 비용 범주 이름은 특수 문자와 공백을 제거하기 위해 정규화됩니다. 정규화 후 충돌하는 태그 또는 비용 범주가 있을 경우 하나의 값만 유지됩니다. 자세한 내용은 열 이름 을 참조하세요. 레거시 CUR CSV: 태그 및 비용 범주 이름은 변경되지 않습니다.

CUR 2.0의 스키마에 대한 자세한 내용은 [데이터 내보내기 테이블 사전](#)을 참조하세요.

다음 두 가지 방법으로 데이터 내보내기에서 CUR 2.0으로 마이그레이션할 수 있습니다.

- [방법 1: CUR 스키마를 사용하여 SQL 쿼리를 사용하여 내보내기 생성](#)
- [방법 2: 새 스키마를 사용하여 CUR 2.0의 내보내기 생성](#)

방법 1: CUR 스키마를 사용하여 SQL 쿼리를 사용하여 내보내기 생성

SQL 쿼리를 사용하여 내보내기를 생성할 수 있습니다. 내보내기 스키마는 현재 CUR로 받는 스키마와 일치합니다. AWS API 또는 SDK를 사용하여이 작업을 수행합니다.

1. 현재 CUR과 일치시키는 데 필요한 (a) 열 목록과 (b) CUR 콘텐츠 설정(리소스 ID 포함, 분할 비용 할당 데이터, 시간 세부 수준)을 결정합니다.
 1. CUR 파일 중 하나의 스키마를 보거나 매니페스트 파일로 이동하여 열 목록을 추출하여 열 목록을 확인할 수 있습니다.

2. 콘솔의 데이터 내보내기로 이동한 다음 CUR 내보내기를 선택하여 세부 정보를 확인하면 CUR 콘텐츠 설정을 확인할 수 있습니다.
2. 이름이 COST_AND_USAGE_REPORT인 CUR 2.0 테이블에서 식별한 열을 선택하는 SQL 쿼리를 작성합니다.

1. CUR 2.0 테이블의 모든 열 이름은 모두 소문자를 사용하고 단어와 단어 사이를 밑줄로 연결하는 '스네이크 케이스'(예: line_item_usage_amount)로 되어 있습니다. SQL 문의 경우 이전 열 이름을 스네이크 케이스로 변환해야 할 수 있습니다.
2. SQL 문의 경우 CUR 2.0에서 중첩된 열을 선택하려면 모든 resource_tag 및 cost_category 열과 특정 product 및 discount 열을 점 연산자가 포함되도록 변환해야 합니다. 예를 들어, CUR 2.0에서 product_from_location 열을 선택하려면 product.from_location을 선택하여 SQL 문을 작성하세요.

예시: `SELECT product.from_location FROM COST_AND_USAGE_REPORT`

그러면 product 맵 열의 from_location 열이 선택됩니다.

3. 기본적으로 점 연산자로 선택한 열의 이름은 속성(예: from_location)에 따라 지정됩니다. 기존 CUR과 일치시키려면 열의 별칭을 이전과 동일하게 선언해야 합니다.

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

중첩된 열에 대한 자세한 내용은 [데이터 내보내기 테이블 사전](#)을 참조하세요.

3. 1단계에서 식별한 CUR 콘텐츠 설정을 CreateExport API의 테이블 구성 형식으로 작성합니다. 다음 단계에서 데이터 쿼리와 함께 이러한 테이블 구성을 제공해야 합니다.
4. 데이터 내보내기용 AWS SDK/CLI에서 CreateExport API를 사용하여 SQL 쿼리 및 테이블 구성을 데이터 쿼리 필드에 입력합니다.
 1. 대상 Amazon S3 버킷 및 덮어쓰기 기본 설정과 같은 전송 기본 설정을 지정합니다. 이전과 동일한 전송 기본 설정을 선택하는 것이 좋습니다. 필수 필드에 대한 정보는 AWS 결제 및 비용 관리 API의 [AWS Data Exports](#) 섹션을 참조하세요.
 2. 데이터 내보내기가 버킷에 쓸 수 있도록 대상 Amazon S3 버킷의 권한을 업데이트합니다. 자세한 내용은 [Amazon S3 버킷에 데이터 내보내기 설정](#)을 참조하세요.
5. CUR 2.0이 전송되는 대상 Amazon S3 버킷의 디렉터리에서 데이터를 읽도록 데이터 모으기 파이프라인에 지시합니다.

방법 2: 새 스키마를 사용하여 CUR 2.0의 내보내기 생성

중첩된 열과 추가 열로 구성된 새 스키마를 사용하여 CUR 2.0의 내보내기를 생성할 수 있습니다. 하지만 이러한 새 열을 처리하려면 현재 데이터 파이프라인을 조정해야 합니다. 콘솔, AWS API 또는 SDK를 사용하여이 작업을 수행합니다.

1. 현재 CUR과 일치시키는 데 필요한 CUR 콘텐츠 설정(리소스 ID 포함, 분할 비용 할당 데이터, 시간 세부 수준)을 결정합니다.
 - 콘솔의 데이터 내보내기로 이동한 다음 CUR 내보내기를 선택하여 세부 정보를 확인하면 CUR 콘텐츠 설정을 확인할 수 있습니다.
2. 데이터 내보내기 콘솔 페이지(옵션 A) 또는 AWS SDK/CLI(옵션 B)를 사용하여 '비용 및 사용량 보고서' 테이블의 모든 열을 선택하는 CUR 2.0 내보내기를 생성합니다.
3. (옵션 A) 콘솔에서 내보내기를 생성하는 방법:

1. 탐색 창에서 데이터 내보내기를 선택합니다.
2. 데이터 내보내기 페이지에서 생성을 선택합니다.
3. 표준 데이터 내보내기를 선택합니다.

비용 및 사용량 보고서(CUR 2.0) 테이블의 경우 기본적으로 모든 열이 선택됩니다.

4. 1단계에서 식별한 CUR 콘텐츠 설정을 지정합니다.
 5. 데이터 테이블 전송 옵션에서 옵션을 선택합니다.
 6. 생성(Create)을 선택합니다.
4. (옵션 B) AWS API/SDK를 사용하여 내보내기를 생성하려면 먼저 COST_AND_USAGE_REPORT 테이블의 모든 열을 선택하는 쿼리를 작성합니다.
 1. GetTable API를 사용하여 전체 열 목록을 확인하고 전체 스키마를 받습니다.
 2. 1단계에서 식별한 CUR 콘텐츠 설정을 CreateExport API의 테이블 구성 형식으로 작성합니다.
 3. CreateExport API를 사용하여 SQL 쿼리와 테이블 구성을 data-query 필드에 입력합니다.
 4. 대상 Amazon S3 버킷 및 덮어쓰기 기본 설정과 같은 전송 기본 설정을 지정합니다. 이전과 동일한 전송 기본 설정을 선택하는 것이 좋습니다. 필수 필드에 대한 정보는 AWS 결제 및 비용 관리 API의 [AWS Data Exports](#) 섹션을 참조하세요.
 5. 데이터 내보내기가 버킷에 쓸 수 있도록 대상 Amazon S3 버킷의 권한을 업데이트합니다. 자세한 내용은 [Amazon S3 버킷에 데이터 내보내기 설정](#)을 참조하세요.

5. CUR 2.0이 전송되는 대상 Amazon S3 버킷의 디렉터리에서 데이터를 읽도록 데이터 모으기 파이프라인에 지시합니다.

또한 중첩된 키값이 있는 새 열(product, resource_tags, cost_category, discounts)을 처리하도록 데이터 모으기 파이프라인과 비즈니스 인텔리전스 도구를 업데이트해야 합니다.

데이터 내보내기 생성

Billing and Cost Management 콘솔의 데이터 내보내기 페이지를 사용하여 표준 내보내기, Cost and Usage Dashboard 내보내기, 레거시 내보내기의 세 가지 유형의 데이터 내보내기를 생성할 수 있습니다.

테이블당 생성할 수 있는 내보내기 수에는 다음과 같은 제한 사항이 적용됩니다.

- 비용 및 사용량 보고서 2.0(CUR 2.0): 내보내기 5개
- 비용 최적화 권장 사항: 내보내기 2개
- AWS 열이 있는 FOCUS 1.0: 내보내기 2개
- AWS 열이 있는 FOCUS 1.2: 내보내기 2개
- Cost and Usage Dashboard: 내보내기 2개
- 탄소 배출량: 내보내기 2개

자세한 내용은 [할당량 및 제한](#)을 참조하세요.

콘솔에서 내보내기를 생성하고 내보낼 테이블을 선택하거나 SDK AWS /CLI에서 내보내기를 생성하고 원하는 데이터 테이블에서 열 선택 및 행 필터의 SQL 쿼리를 정의하여 몇 분 안에 내보내기를 설정합니다.

콘솔에서 내보내기를 생성할 때 데이터 내보내기 스토리지용 Amazon S3 버킷을 생성할 수 있습니다. AWS SDK/CLI에서 내보내기를 생성할 때는 올바른 버킷 정책을 사용하여 Amazon S3 버킷을 미리 생성해야 합니다. 자세한 내용은 [Amazon S3 버킷에 데이터 내보내기 설정](#)을 참조하세요.

새 데이터 내보내기를 생성하면 데이터 내보내기가 데이터를 Amazon S3 버킷으로 내보내기 시작합니다.

Note

AWS 가 Amazon S3 버킷으로 내보내기 전송을 시작하는 데 최대 24시간이 걸릴 수 있습니다. 전송이 시작되면 S3 버킷에서 결제 및 비용 관리 내보내기 출력을 하루에 한 번 이상 새로 고치고 탄소 배출량 내보내기 출력을 한 달에 한 번 이상 AWS 새로 고칩니다. 실제 새로 고침 빈도는 다양한 요인으로 인해 달라질 수 있습니다.

- [Amazon S3 버킷에 데이터 내보내기 설정](#)
- [표준 내보내기 생성](#)
- [Cost and Usage Dashboard 생성](#)
- [레거시 CUR 내보내기 생성](#)
- [결제 보기를 사용하여 내보내기 생성](#)
- [데이터 쿼리-SQL 쿼리 및 테이블 구성](#)
- [AWS Billing Conductor를 사용하여 비용 및 사용 보고서 2.0 구성](#)

Amazon S3 버킷에 데이터 내보내기 설정

데이터 내보내기를 수신하고 저장하려면 계정 또는 지정된 대상 AWS 계정에 Amazon S3 버킷이 있어야 합니다 AWS . 콘솔에서 내보내기를 생성할 때 자체 버킷에서 내보내기를 원하는 경우 소유한 기존 S3 버킷을 선택하거나 새 버킷을 생성할 수 있습니다. 어느 경우든 다음 기본 S3 버킷 정책의 적용을 검토 및 확인해야 합니다. 내보내기를 다른 AWS 계정이 소유한 버킷으로 전송하려면 데이터 내보내기 생성 프로세스 중에 버킷 소유자와 버킷 이름을 지정할 수 있습니다. 내보내기를 생성한 후 버킷 정책을 편집하거나 S3 버킷 소유자를 변경하면 Data Exports가 내보내기를 전송하지 못할 수 있습니다. 모든 S3 버킷에 내보내기 데이터를 저장하면 표준 Amazon S3 요금이 청구됩니다. 자세한 내용은 [할당량 및 제한](#)을 참조하세요.

데이터 내보내기를 생성할 때 사용자 또는 다른 AWS 계정이 소유한 모든 S3 버킷에 다음 정책을 적용해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnableAWSDataExportsToWriteToS3",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "bcm-data-exports.amazonaws.com"
        ]
      },
      "Action": [
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::{bucket-name}/*",
      "Condition": {
```

```

    "ArnLike": {
      "aws:SourceArn": "arn:aws:bcm-data-exports:us-east-1:{source-account-id}:export/*"
    },
    "StringEquals": {
      "aws:SourceAccount": "{source-account-id}"
    }
  }
}
]
}

```

이 S3 버킷 정책은 데이터 내보내기가 내보내기를 생성한 계정을 대신하여 S3 버킷으로만 내보내기를 전송할 수 있도록 합니다. 또한 Data Exports는 내보내기 생성 중에 지정된 계정이 S3 버킷을 여전히 소유하고 있는지 확인할 수 있습니다.

- S3 버킷으로 내보내기를 전송하려면 해당 S3 버킷에 대한 쓰기 권한이 AWS 필요합니다. 이를 위해 S3 버킷 정책은 소유한 S3 버킷(arn:aws:s3:::<EXAMPLE-BUCKET>/*)에 보고서를 전달 (s3:PutObject)할 수 있는 서비스(bcm-data-exports.amazonaws.com) 권한을 부여합니다.
- 데이터 내보내기에서 S3 버킷에 쓰기 요청을 할 때마다 내보내기를 생성한 계정의 계정 ID를 제공해야 합니다. aws:SourceArn 및 aws:SourceAccount 조건 키가 이를 적용합니다.
- 이 S3 버킷 정책은 전송된 비용 및 사용 보고서를 포함하여 S3 버킷의 객체를 읽거나 삭제할 수 있는 AWS 권한을 부여하지 않습니다.

액세스 제어 목록(ACL)이 활성화된 Amazon S3 버킷의 경우 데이터 내보내기는 보고서를 전송할 때 보고서에 BucketOwnerFullControl ACL을 추가로 적용합니다. 기본적으로 이러한 보고서와 같은 Amazon S3 객체는 해당 객체를 작성한 사용자 또는 서비스 주체만 읽을 수 있습니다. 사용자 또는 S3 버킷 소유자에게 보고서를 읽을 수 있는 권한을 제공하려면 AWS가 BucketOwnerFullControl ACL을 적용해야 합니다. ACL은 S3 버킷 소유자에게 이 보고서에 대한 Permission.FullControl 권한을 부여합니다. 하지만 ACL을 비활성화하고 S3 버킷 정책을 사용하여 액세스를 제어하는 것이 좋습니다.

Note

새로 만든 S3 버킷의 경우 기본적으로 ACL이 비활성화됩니다. 자세한 내용은 [객체 소유권 제어 및 버킷에 대해 ACL 사용 중지](#)를 참조하십시오.

데이터 내보내기 콘솔 페이지에 잘못된 버킷 오류가 표시되는 경우 보고서 설정 이후 정책 및 S3 버킷 소유권이 변경되지 않았는지 확인하세요.

표준 내보내기 생성

다른 처리 도구(예: Amazon Athena)를 사용하여 분석할 수 있는 표준 데이터 내보내기를 생성할 수 있습니다.

표준 데이터 내보내기를 생성하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 내보내기 생성을 선택합니다.
4. 내보내기 생성 페이지의 내보내기 유형에서 표준 데이터 내보내기를 선택합니다.
5. 내보내기 이름에 내보내기의 이름을 입력합니다.

내보내기 이름은 최대 128자일 수 있으며 고유해야 합니다. 유효한 문자는 a~z, A~Z, 0~9, -(하이픈) 및 _(밑줄)입니다.

6. 데이터 테이블 설정에서 내보내기에 포함할 테이블과 열을 지정할 수 있습니다. 먼저 내보낼 테이블을 선택합니다.

Note

비용 최적화 권장 사항 테이블을 내보내려면 서비스 연결 역할이 필요합니다. 자세한 내용은 [데이터 내보내기의 서비스 연결 역할](#)을 참조하세요.

탄소 배출량 테이블을 내보내려면 탄소 발자국 데이터에 액세스할 `sustainability:GetCarbonFootprintSummary` 수 있는 IAM 권한이 필요합니다.

AWS 열이 있는 FOCUS 1.0과 탄소 배출량을 제외하고 내보내기에 데이터를 추가하기 위한 다양한 테이블 구성이 있습니다.

1. CUR 2.0:

- a. 내보내기에 각 개별 리소스의 ID를 포함하려면 리소스 ID 포함을 선택합니다.

Note

리소스 ID를 포함하면 각 리소스에 대한 개별 항목이 생성됩니다. 이렇게 하면 AWS 사용량에 따라 내보내기 크기가 크게 증가할 수 있습니다.

리소스 ID를 선택하면 새 보고서를 생성할 때 사용자, 계정, 비용 범주 및 리소스에 대한 데이터가 포함된 태그 열이 추가됩니다. 중복 정보를 방지하기 위해 열을 선택 취소할 수 있습니다.

- b. 공유 리소스에 대한 자세한 비용 및 사용량을 포함하려면 비용 할당 데이터 분할을 선택합니다(Amazon ECS 및 Amazon EKS).

Note

비용 할당 데이터 분할을 포함하면 각 리소스(즉, ECS 작업 및 Kubernetes 포드)에 대한 개별 항목이 생성됩니다. 이로 인해 AWS 사용량에 따라 비용 및 사용량 보고서 크기가 크게 늘어날 수 있습니다.

- c. 용량 예약 데이터 포함을 선택하여 내보내기에 용량 예약 열과 행 수준 세부 수준을 포함합니다.

Note

용량 예약 데이터를 포함하면 3개의 새 열이 생성되고 AWS 사용량에 따라 인스턴스 행 항목을 분할할 수 있습니다.

- d. 비용 및 사용량 보고서에 표준 자동 할인 형식 대신 수동 할인 형식으로 표시되도록 할인을 변환하려면 수동 할인 형식 활성화를 선택합니다.


Note

이 옵션은 할인 자동화 프로그램을 사용하는 경우에만 나타납니다.

- e. 시간 세분 수준에서 시간별, 일별 또는 월별 중에서 선택하여 내보내기의 항목을 해당 시간 세분화별로 집계합니다.


2. AWS 열이 있는 FOCUS의 경우 테이블 구성이 없습니다.
3. 탄소 배출량의 경우 테이블 구성이 없습니다.
4. 비용 최적화 권장 사항:

- a. 모든 권장 사항 포함을 선택하여 서로 호환되지 않는 권장 사항의 절감 가치가 가장 낮은 권장 사항을 제거합니다.
- b. 호환되지 않는 권장 사항을 제거하기 전에 특정 유형의 권장 사항을 필터링하려면 권장 사항 필터를 추가합니다.

 Note

Cost Optimization Hub 콘솔에서 해당 설정을 지정한 경우 Cost Optimization Hub에서 내보내기 생성을 선택하면 데이터 내보내기로 전달됩니다.


7. 열 선택에서 내보내기에 포함할 열을 선택합니다. 확실하지 않은 경우 테이블 상단의 첫 번째 확인란을 선택하여 모든 열을 선택하세요. 열을 더 선택하면 내보내기 파일 크기가 커질 수 있습니다.
8. 데이터 테이블 전송 옵션에서 데이터 내보내기 새로 고침 주기:
 - 결제 및 비용 관리 데이터 내보내기의 경우 사용 가능한 유일한 옵션은 일별 - 내보내기는 최대 1일 1회 갱신입니다.
 - 탄소 배출량 데이터 내보내기의 경우 사용 가능한 유일한 옵션은 월별 - 내보내기는 월 1회 갱신입니다. 각 업데이트는 이전 달의 탄소 배출량 데이터를 제공합니다(예: 2월 업데이트에는 1월 데이터가 포함됨).
9. 압축 유형 및 파일 형식의 경우 내보내기에 다음 중 하나를 선택하세요.
 - Parquet – Parquet
 - gzip – text/csv
10. 파일 버전 관리의 경우 다음 중 하나를 선택하여 업데이트할 때마다 내보내기를 덮어쓸지를 결정하세요.
 - 기존 데이터 내보내기 파일 덮어쓰기: 각 내보내기 새로 고침은 데이터 파티션(예: 청구 기간) 내의 이전 전송을 덮어씁니다. 내보내기를 덮어쓰면 Amazon S3 스토리지 비용을 절약할 수 있습니다.

 Note

비용 최적화 권장 사항 내보내기에는 덮어쓰기가 지원되지 않습니다.


- 새 데이터 내보내기 파일 생성: 동일한 파티션(예: 청구 기간)에 전송되었더라도 각 내보내기 새로 고침이 별도의 디렉터리에 기록됩니다. 새 내보내기 버전을 만들면 시간 경과에 따른 비용 및 사용량 데이터의 변화를 추적할 수 있습니다.

11. 데이터 내보내기 스토리지 설정에서 내보내기를 다음 S3 버킷으로 전송할지 여부를 선택합니다.
 - 이 계정
 - 다른 계정
12. S3 버킷 이름에 대해 이 계정을 선택하는 경우 구성을 선택하고 다음 중 하나를 수행합니다.
 - 기존 버킷을 선택합니다.
 - 버킷 생성을 선택하고 S3 버킷 이름을 입력한 다음 새 버킷을 생성할 리전을 선택합니다.
 - 버킷 정책을 검토합니다. 기존 버킷을 선택하는 경우 Data Exports가 기존 S3 버킷 정책을 덮어 쓸 것임을 확인해야 합니다. 새 정책에서는 CUR 및 데이터 내보내기 모두 내보내기를 전송할 수 있게 됩니다.
13. 다른 계정을 선택한 경우 S3 버킷, 버킷을 소유한 계정인 S3 버킷 소유자 및 리전을 입력합니다.
AWS
14. S3 경로 접두사의 경우 모든 내보내기 데이터를 저장하기 위해 S3 버킷에 생성할 디렉터리의 이름을 입력합니다.

 Note

내보내기가 다른 계정의 S3 버킷으로 전송되는 경우 동일한 경로 접두사를 가진 여러 계정과 보고서 이름이 실수로 서로의 데이터를 덮어쓰지 않도록 계정에 고유한 S3 경로 접두사를 사용하는 것이 좋습니다.

15. 태그에서 최대 50개의 태그를 추가하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

 Note

태그 추가는 선택 사항입니다.

16. 생성을 선택하여 내보내기 생성을 완료합니다.

Cost and Usage Dashboard 생성

Amazon QuickSight에서 제공하는 사전 구축된 Cost and Usage Dashboard를 배포하여 청구 및 비용 관리 데이터를 시각화할 수 있습니다.

Cost and Usage Dashboard를 생성하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 데이터 내보내기 페이지에서 생성 또는 Cost and Usage Dashboard 타일을 선택합니다.
4. 생성 페이지의 내보내기 유형에서 QuickSight에서 제공하는 Cost and Usage Dashboard를 선택합니다.
5. 내보내기 이름에 대시보드의 이름을 입력합니다.

내보내기 이름은 최대 128자일 수 있으며 고유해야 합니다. 유효한 문자는 a~z, A~Z, 0~9, -(하이픈) 및 _(밑줄)입니다.


6. QuickSight 대시보드 설정의 경우 계정 이름, 계정 ID, 계정 버전 및 인증 방법과 같은 QuickSight 계정 세부 정보가 자동으로 채워집니다.
 1. QuickSight 계정 세부 정보가 자동으로 채워지지 않는다면 QuickSight를 처음 사용하는 경우 계정 생성을 선택하여 가입하고, 기존 QuickSight 고객인 경우 QuickSight 계정에 로그인하세요.
 2. QuickSight 계정을 성공적으로 만들거나 로그인하면 성공 메시지가 표시됩니다. 창을 닫고 데이터 내보내기로 돌아갑니다.
 3. QuickSight 대시보드 설정에서 새로 고침을 선택합니다.

Note

이 기능을 사용하려면 [Enterprise Edition](#)이 필요합니다.

7. QuickSight 네임스페이스에 [네임스페이스](#)를 입력합니다.
8. QuickSight 사용자 이름에 QuickSight 대시보드에 액세스할 권한이 있는 사용자의 세부 정보를 입력합니다.
9. QuickSight 리전에서 QuickSight 대시보드를 생성할 AWS 리전을 선택합니다.
10. 데이터 테이블 콘텐츠 설정 및 데이터 테이블 전송 옵션은 사전 설정되어 있으며 편집할 수 없습니다.
11. 데이터 내보내기 스토리지 설정의 S3 버킷 이름에서 구성을 선택합니다.
12. S3 버킷 구성 대화 상자에서 다음 중 하나를 수행합니다.
 - 기존 버킷을 선택합니다.
 - 버킷 생성을 선택하고 S3 버킷 이름을 입력한 후 새 버킷을 생성할 리전을 선택합니다.

13. 버킷 정책을 검토한 다음 버킷 생성을 선택합니다.
14. S3 경로 접두사에 내보내기 이름의 앞에 추가할 S3 경로 접두사를 입력합니다.
15. 서비스 액세스에서 QuickSight에 권한을 부여할 방법을 선택합니다.
 - 새 서비스 역할 생성(기본값)
 - 기존 서비스 역할 사용
16. 태그에서 최대 50개의 태그를 추가하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

 Note

태그 추가는 선택 사항입니다.

17. 생성(Create)을 선택합니다.

언제든지 AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지로 돌아가 비용 및 사용 대시보드가 마지막으로 업데이트된 시기를 확인할 수 있습니다.

레거시 CUR 내보내기 생성

레거시 비용 및 사용량 보고서(CUR)의 데이터 내보내기를 생성할 수 있습니다. 이 워크플로는 레거시 cur API를 사용하며 SQL을 사용하여 내보내기 콘텐츠를 정의할 수 없습니다. 추가 열과 SQL 액세스가 포함된 CUR 2.0은 표준 데이터 내보내기로만 사용할 수 있습니다.

레거시 데이터 내보내기를 생성하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 생성(Create)을 선택합니다.
4. 생성 페이지의 내보내기 유형에서 레거시 CUR 내보내기를 선택합니다.
5. 내보내기 이름에 내보내기의 이름을 입력합니다.
6. 콘텐츠 내보내기에서 CUR 내보내기에 포함할 데이터를 선택합니다.
 - 추가 내보내기 콘텐츠에서 리소스 ID 포함을 선택하여 보고서에 각 개별 리소스의 ID를 포함합니다.

Note

리소스 ID를 포함하면 각 리소스에 대한 개별 항목이 생성됩니다. 이로 인해 AWS 사용량에 따라 내보내기 파일 크기가 크게 늘어날 수 있습니다.

- 공유 리소스에 대한 자세한 비용 및 사용량을 포함하려면 비용 할당 데이터 분할을 선택합니다 (Amazon ECS 및 Amazon EKS).

Note

비용 할당 데이터 분할을 포함하면 각 리소스(즉, ECS 작업 및 Kubernetes 포드)에 대한 개별 항목이 생성됩니다. 이렇게 하면 AWS 사용량에 따라 비용 및 사용 보고서의 크기가 크게 증가할 수 있습니다.

- 비용 및 사용량 보고서에 표준 자동 할인 형식 대신 수동 할인 형식으로 표시되도록 할인을 변환하려면 수동 할인 형식 활성화를 선택합니다.

Note

이 기능은 할인 자동화 이용 고객만 사용할 수 있습니다.

7. 데이터 테이블 전송 옵션의 시간 세부 수준에서 다음 중 하나를 선택합니다.


- 시간별: 내보내기의 품목을 시간별로 집계하려면 선택합니다.
- 일별: 내보내기의 품목을 일별로 집계하려면 선택합니다.
- 월별: 내보내기의 품목을 월별로 집계하려면 선택합니다.

8. 보고서 버전 관리의 경우 다음 중 하나를 선택합니다.

- 새 보고서 버전 생성: 청구 기간이 같은 전송이라도 보고서를 새로 고칠 때마다 별도의 디렉터리에 기록됩니다. 시간 경과에 따른 내보내기를 감사하는 기능을 개선하려면 이 옵션을 선택하세요.
- 기존 보고서 덮어쓰기: 보고서를 새로 고칠 때마다 동일한 청구 기간 내의 이전 전송을 덮어씁니다. 새 청구 기간의 전송은 새 파일 및 디렉터리로 전달됩니다. Amazon S3 스토리지 비용을 절약하려면 이 옵션을 선택하세요.

9. 보고서 데이터 통합에서 비용 및 사용량 보고서를 Amazon Athena, Amazon Redshift 또는 Amazon QuickSight와 통합하도록 활성화할지를 선택합니다. 보고서는 다음 형식으로 압축됩니다.

- Amazon Athena: Amazon Athena에 가장 적합한 Parquet 파일 형식의 전송 옵션을 선택하고 기존 보고서를 덮어씁니다. 또한 통합을 설정하는 데 사용할 수 있는 스크립트도 제공합니다.
 - Amazon Redshift: Amazon Redshift에 가장 적합한 gzip/csv 파일 형식의 전송 옵션을 선택합니다. 또한 통합을 설정하는 데 사용할 수 있는 스크립트도 제공합니다.
 - Amazon QuickSight: Amazon QuickSight에 가장 적합한 gzip/csv 파일 형식의 전송 옵션을 선택합니다.
10. 압축 유형 및 파일 형식의 경우 다음 중에서 선택하세요.
- Parquet – Parquet
 - gzip – text/csv
 - zip – text/csv
11. 데이터 내보내기 스토리지 설정의 S3 버킷 이름에서 구성을 선택합니다.
12. S3 버킷 구성 대화 상자에서 다음 중 하나를 수행합니다.
- 기존 버킷을 선택합니다.
 - 버킷 생성을 선택하고 S3 버킷 이름을 입력한 후 새 버킷을 생성할 리전을 선택합니다.
13. 버킷 정책을 검토한 다음 버킷 생성을 선택합니다.
14. S3 경로 접두사에 내보내기 이름의 앞에 추가할 S3 경로 접두사를 입력합니다.
15. 태그에서 최대 50개의 태그를 추가하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

 Note

태그 추가는 선택 사항입니다.

16. 보고서 생성을 선택합니다.

결제 보기를 사용하여 내보내기 생성

결제 전송을 사용하여 청구서 전송 계정으로 로그인하거나 AWS Billing Conductor를 사용하여 관리 계정으로 로그인하면 AWS 관리형 결제 보기(결제 그룹 및 결제 전송 보기)를 기반으로 내보내기를 생성할 수 있습니다.

⚠ Important

- 사용자 지정 결제 보기는 지원되지 않습니다.
- 데이터 내보내기 페이지에서만 결제 보기 기반 보고서를 생성할 수 있습니다. 레거시 비용 및 사용 보고서 페이지에서는 결제 보기를 기반으로 보고서 생성을 지원하지 않습니다.

보고서는 계정의 리소스이므로 결제 보기 모드의 활성화 또는 비활성화 여부에 관계없이 결제 보기를 기반으로 보고서를 생성할 수 있습니다.

결제 보기를 기반으로 보고서를 생성하려면

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 보고서 생성을 선택합니다.
4. 결제 보기 유형을 선택합니다(관리형 보기만 해당).
5. 보고서의 특정 보기를 선택합니다.
6. 나머지 단계를 완료하여 보고서를 생성합니다.

ℹ Note

결제 전송 쇼백/ 차지백 보기 또는 결제 그룹 보기를 기반으로 보고서를 생성할 때는 비용 할당 데이터 분할 기능을 비활성화해야 합니다.

결제 전송 사용 사례를 위한 데이터 내보내기에 대한 자세한 내용은 [결제 전송 모범 사례](#)를 참조하세요.

데이터 쿼리-SQL 쿼리 및 테이블 구성

데이터 내보내기를 사용하면 제공된 테이블(예: CUR 2.0)에 대해 실행되는 SQL 쿼리(열 선택, 행 필터링, 열 별칭 지정)를 작성할 수 있습니다. 또한 각 테이블에는 테이블에 포함된 데이터를 변경하는 테이블 구성이 있을 수 있습니다. 예를 들어 CUR 2.0에서는 시간별, 일별 또는 월별로 시간 세부 수준을 선택하는 구성을 지정하거나 리소스 수준의 세분성으로 비용 및 사용량 데이터를 추가하는 구성을 지정할 수 있습니다.

데이터 내보내기 쿼리를 완전히 정의하려면 다음 두 속성을 지정해야 합니다.

- SQL 쿼리: SQL 문은 테이블에 대해 실행되며 내보내기에서 반환되는 데이터를 결정합니다.
- 테이블 구성: 테이블 구성 설정은 테이블에 대해 SQL 쿼리를 실행하기 전에 테이블에 포함된 데이터를 변경합니다.

데이터 내보내기 콘솔 페이지에서 선택 사항에 따라 SQL 문 및 테이블 구성을 작성하는 워크플로를 사용할 수 있습니다. 데이터 내보내기 SDK/CLI에서 자체 SQL 문 및 테이블 구성을 작성할 수 있습니다.

데이터 내보내기 SQL 문(QueryStatement)은 다음 구문을 사용합니다.

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

데이터 내보내기 테이블 구성(TableConfigurations)은 다음 구문을 사용합니다.

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

SQL 쿼리

SQL 쿼리는 테이블에 대해 실행되며 내보내기에 반환되는 데이터를 결정합니다. 내보내기를 만든 후 SQL 문을 변경할 수 있지만 선택한 테이블은 변경할 수 없습니다.

SQL 문(QueryStatement 필드)은 최대 3만 6,000자까지 입력할 수 있습니다.

데이터 내보내기 SQL 쿼리에 사용할 수 있는 키워드는 다음과 같습니다.

Note

키워드는 대/소문자를 구분하지 않습니다. 열 이름과 테이블 이름은 대/소문자를 구분합니다.

SELECT

필수 사항입니다.

테이블에서 선택할 열을 지정합니다. 쿼리당 SELECT 문을 하나만 사용할 수 있습니다.

점 연산자(.)를 사용하여 MAP 또는 STRUCT 열의 속성 선택을 별도의 열로 지정하세요. SQL 출력의 결과 열 이름은 기본적으로 속성 이름입니다.

예를 들어 제품 MAP 열에서 속성을 선택할 수 있습니다.

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

그러면 product 열에서 from_location 속성이 선택되고 속성 데이터가 포함된 새 열이 생성됩니다. 기본적으로 출력에서 이 열의 이름은 from_location이 됩니다. 하지만 AS를 사용하여 이름을 바꿀 수 있습니다.

각 테이블에서 사용할 수 있는 MAP 및 STRUCT 열과 이러한 열의 속성에 대한 자세한 내용은 [데이터 내보내기 테이블 사전](#)을 참조하세요.

AS

선택 사항.

선택 중인 열의 이름을 바꿀 수 있게 해줍니다. 새 이름 열에는 공백이 있어서는 안 되며 영숫자(a~z, A~Z, 0~9)와 밑줄(_)만 사용해야 합니다. 다른 문자를 사용하기 위해 열 별칭을 정의할 때는 따옴표를 사용할 수 없습니다.

별칭 지정은 MAP 또는 STRUCT 열의 속성을 선택하여 결과 열의 이름을 CUR의 스키마와 일치하도록 바꿀 때 유용할 수 있습니다. 예를 들어 CUR에서 product_from_location 열을 표시하는 방식과 일치시키려면 CUR 2.0 테이블을 사용하여 데이터 내보내기에 다음 쿼리를 작성하세요.

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

그러면 product_from_location이라는 열이 있는 내보내기가 생성됩니다.

FROM

필수 사항입니다.

쿼리할 테이블을 지정합니다. 쿼리당 FROM 문을 하나만 사용할 수 있습니다.

WHERE

선택 사항.

행을 필터링하여 지정된 절과 일치하는 행만 표시합니다.

WHERE 절은 다음과 같은 연산자를 지원합니다.

- = 값이 문자열 또는 숫자와 일치해야 합니다.
- != and <> 값이 지정된 문자열 또는 숫자와 일치하지 않아야 합니다.
- <, <=, >, 및 >= 값이 숫자보다 작거나, 작거나 같거나, 크거나, 크거나 같아야 합니다.
- AND. 지정된 두 조건과 모두 일치하려면 true여야 합니다. 여러 AND 키워드를 사용하여 2개 이상의 조건을 지정할 수 있습니다.
- OR 지정된 두 조건 중 하나가 true여야 일치합니다. 여러 OR 키워드를 사용하여 2개 이상의 조건을 지정할 수 있습니다.
- NOT 지정된 조건이 true가 아니어야 일치합니다.
- IN 키워드 뒤의 괄호 안에 지정된 모든 값이 true여야 일치합니다.
- 괄호를 사용하여 다중 조건부 WHERE 절을 생성할 수 있습니다.

Note

문자열을 연산자 뒤에 오는 값으로 표현할 때는 큰따옴표 대신 작은따옴표(')를 사용하세요. 작은따옴표를 이스케이프 처리할 필요는 없습니다. 예를 들어 다음과 같은 WHERE 문을 작성할 수 있습니다.

```
WHERE line_item_type = 'Discount' OR line_item_type = 'Usage'
```

LIMIT

선택 사항.

쿼리에 의해 반환되는 행 수를 지정한 값으로 제한합니다.

테이블 구성

테이블 구성은 데이터 내보내기에서 테이블을 쿼리하기 전에 사용자가 테이블의 데이터나 스키마를 변경하도록 설정할 수 있는 사용자 제어 속성입니다. 테이블 구성은 JSON 문으로 저장되며 AWS SDK/CLI의 사용자 입력 또는 콘솔의 사용자 선택을 통해 지정됩니다.

예를 들어 CUR 2.0에는 데이터 세부 수준(시간별, 일별, 월별), 리소스 수준의 세분화된 데이터 포함 여부, 분할 비용 할당 데이터 포함 여부 등을 변경할 수 있는 테이블 구성이 있습니다. 모든 테이블에 구성

이 있는 것은 아닙니다. 각 테이블에 사용할 수 있는 구성에 대한 자세한 내용은 [데이터 내보내기 테이블 사전](#)을 참조하세요.

각 테이블 구성 파라미터에는 사용자가 테이블 구성을 지정하지 않은 경우 가정되는 기본값이 있습니다. 내보내기가 생성된 후 테이블 구성을 변경할 수 없습니다.

AWS Billing Conductor를 사용하여 비용 및 사용 보고서 2.0 구성

AWS Billing Conductor를 사용하면 각 결제 그룹에 대해 pro forma AWS Cost and Usage Report (AWS CUR) 2.0을 생성할 수 있습니다. 이러한 견적 보고서는 표준 AWS CUR 2.0과 동일한 파일 형식, 세분화 및 열을 사용하여 지정된 기간 동안 사용 가능한 가장 포괄적인 비용 및 사용 데이터를 제공합니다.

AWS Billing Conductor에 대한 자세한 내용은 [AWS Billing Conductor 사용 설명서](#)를 참조하세요.

주제

- [표준 및 AWS Billing Conductor 비용 및 사용 보고서 비교](#)
- [결제 그룹에 대한 견적 비용 및 사용 보고서 생성](#)

표준 및 AWS Billing Conductor 비용 및 사용 보고서 비교

AWS Billing Conductor 구성을 사용하여 생성된 표준 비용 및 사용 보고서와 견적 AWS CUR에는 몇 가지 차이점이 있습니다.

계정 적용 범위

- 표준 AWS CUR - 통합 결제 패밀리리의 모든 계정에 대한 비용 및 사용 데이터를 포함합니다.
- 견적 AWS CUR - 보고서 생성 시 특정 결제 그룹에 속한 계정만 포함

인보이스 처리

- 표준 AWS CUR -가 인보이스를 AWS 생성한 후 인보이스 열을 채웁니다.
- 견적 AWS CUR -는 견적 결제 데이터를 기반으로 인보이스를 생성하거나 발행하지 않으므로 인보이스 열을 채우지 않습니다.

결제 그룹에 대한 견적 비용 및 사용 보고서 생성

다음 단계에 따라 결제 그룹에 대한 견적 AWS CUR을 생성합니다.

결제 그룹을 위한 견적 비용 및 사용 보고서 생성

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 생성(Create)을 선택합니다.
4. 세부 정보 내보내기 섹션에서 표준 데이터 내보내기를 선택합니다.
5. 내보내기 이름에 내보내기의 이름을 입력합니다.
6. 데이터 테이블 콘텐츠 설정에서 CUR 2.0을 선택합니다.
7. 데이터 테이블 구성에서 리소스 IDs 선택하여 보고서에 각 개별 리소스의 IDs를 포함합니다.

견적 데이터 내보내기가 활성화되면 비용 할당 데이터 분할이 비활성화됩니다.

8. 다음을 선택합니다.
9. S3 버킷에서 구성을 선택합니다.
10. S3 버킷 구성 대화 상자에서 다음 중 하나를 수행합니다.
 - 드롭다운 목록에서 기존 버킷을 선택하고 다음을 선택합니다.
 - 버킷 이름과 새 버킷을 생성할 AWS 리전 를 입력하고 다음을 선택합니다.
11. 버킷 정책을 검토하고 이 정책이 올바른지 확인했음을 선택한 다음 저장을 선택합니다.
12. S3 경로 접두사에 내보내기 이름의 앞에 추가할 S3 경로 접두사를 입력합니다.
13. 시간 세부 수준에 대해 다음 중 하나를 선택합니다.
 - 시간별: 보고서의 항목을 시간별로 집계하려면 선택합니다.
 - 일별: 보고서의 항목을 일별로 집계하려면 선택합니다.
 - 보고서의 행 항목을 월별로 집계하려는 경우 매월.
14. 보고서 버전 관리에서, 보고서의 각 버전을 이전 버전을 덮어쓰지 또는 이전 버전과 별도로 추가 제공할지를 선택합니다.

보고서를 덮어쓰면 Amazon S3 스토리지 비용을 절약할 수 있습니다. 새 보고서 버전을 제공하면 시간이 지남에 따라 결제 데이터의 감사 가능성을 개선할 수 있습니다.
15. 다음을 선택합니다.
16. 보고서 설정을 검토했으면 [Review and Complete]를 선택합니다.

데이터 내보내기 보기 및 관리

내보내기에 대한 세부 정보를 보려면 AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지를 사용하세요. 내보내기 파일을 보려면 데이터 내보내기 페이지에서 Amazon S3 버킷의 S3 콘솔 링크를 사용하세요. 내보내기 대시보드를 보려면 데이터 내보내기 페이지의 QuickSight 링크를 사용하거나 QuickSight 콘솔로 직접 이동하여 대시보드를 찾으세요.

내보내기 세부 정보, 파일 및 대시보드를 보는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 내보내기 및 대시보드 목록에서 보려는 내보내기의 이름을 찾습니다.
4. 내보내기 세부 정보를 보려면 내보내기 이름 옆의 링크를 선택하여 내보내기 설정을 설명하는 요약 페이지를 확인합니다.
5. 내보내기 파일을 보려면 S3 버킷 열에서 버킷의 S3 콘솔로 이동하는 링크를 선택합니다.
6. QuickSight 대시보드를 보려면 내보내기 유형 열에서 Cost and Usage Dashboard 링크를 선택합니다.

다음은 내보내기 및 대시보드 목록에 있는 열의 개요입니다.

- 내보내기 이름: 내보내기를 생성할 때 선택한 이름입니다.
- 상태: 내보내기의 상태입니다. 값은 두 가지입니다.
 - 정상: 이 상태는 가장 최근의 내보내기 전송이 성공했음을 나타냅니다.

Note

모든 데이터가 대시보드에 입력되는 데 최대 24시간이 걸릴 수 있으므로 Cost and Usage Dashboard에 이번 달의 데이터가 누락되었을 수 있습니다. 내보내기 상태가 '정상'으로 표시되는 경우 대시보드가 이번 달의 데이터로 업데이트될 때까지 24시간이 걸릴 수 있습니다.

결제 전송을 청구서 전송 계정으로 사용하거나 AWS Billing Conductor를 관리 계정으로 사용하는 경우 결제 보기 모드가 비활성화된 모든 결제 보기 기반 내보내기를 볼 수 있습니다. 결제 보기 모드를 활성화하면 선택한 결제 보기와 연결된 내보내기만 볼 수 있습니다.

- 정상: 이 상태는 가장 최근의 내보내기 전송이 실패했음을 나타냅니다.
- 내보내기 유형: 생성된 내보내기 유형입니다. 데이터 내보내기에는 세 가지 유형의 내보내기가 있습니다.
- 표준 데이터 내보내기: Amazon S3에 반복적으로 전송되는 테이블의 사용자 지정 내보내기입니다.
- Cost and Usage Dashboard: Amazon QuickSight로 내보내고 통합하여 사전 구축된 Cost and Usage Dashboard를 배포합니다. 이것이 대시보드로 연결되는 링크가 됩니다.
- 레거시 CUR 내보내기: 레거시 비용 및 사용량 보고서(CUR)의 내보내기입니다.
- 데이터 테이블: 내보내기에서 쿼리하는 테이블입니다.
- 생성된 날짜: 내보내기가 생성된 시간과 날짜입니다.
- 마지막으로 새로 고친 날짜: 내보내기를 마지막으로 새로 고친 시간과 날짜입니다.
- S3 버킷: 내보내기가 전송되는 대상 S3 버킷입니다.

내보내기 전송에 대한 이해

다음 섹션에서는 내보내기 전송에 대한 정보를 찾을 수 있습니다.

- 내보내기 S3 상위 디렉터리 구조: 내보내기가 전송되는 S3 디렉터리에서 내보내기 데이터가 구조화 되는 방식입니다.
- 내보내기 새로 고침: 내보내기가 S3 디렉터리에서 업데이트되는 빈도입니다.
- 내보내기 덮어쓰기 및 새로 생성: 덮어쓰기로 인해 내보내기 전송이 변경되는 방식과 새 전송 기본 설정이 생성되는 방식입니다.
- 내보내기 데이터 파일 이름 및 체크: 내보내기 파일(gzip/csv 또는 Parquet)의 이름 지정 방식입니다.

내보내기 S3 상위 디렉터리 구조

각 내보내기는 S3에 대한 쿼리의 데이터(하나 이상의 gzip/csv 또는 Parquet 파일)와 내보내기가 실행된 시점의 내보내기 정의에 대한 정보가 들어 있는 Manifest.json 메타데이터 파일을 S3로 전달합니다.

데이터

내보내기 쿼리의 결과 데이터는 다음 S3 파일 경로에 저장됩니다.

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

파티션은 쿼리 중인 테이블에 해당합니다. CUR 2.0의 경우 파티션은 해당 CUR 2.0 내보내기의 '청구 기간'에 해당합니다.

prefix: 내보내기에 할당하는 S3 파일 접두사입니다.

export-name: 내보내기에 할당하는 이름입니다.

partition: 파티션은 전송을 위해 단일 테이블을 별도의 테이블로 분할하는 방법을 설명합니다. CUR 2.0의 경우 파티션은 BILLING_PERIOD=YYYY-MM 형식의 '청구 기간'에 해당합니다. 예를 들어, 2023년 11월의 파티션은 2023-11입니다.

다음은 S3 파일 경로의 예시입니다:

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

Metadata

쿼리의 Manifest.json 메타데이터 파일은 다음 S3 파일 경로에 저장됩니다.

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

내보내기를 새로 고칠 때마다 Manifest.json 파일이 업데이트됩니다. 내보내기를 통해 생성되는 새 파티션마다 새 Manifest.json 파일이 생성됩니다. CUR 2.0의 경우 이는 새 청구 기간이 시작될 때 새 Manifest.json 파일이 생성됨을 의미합니다.

매니페스트 파일에는 다음과 같은 정보가 포함되어 있습니다.

- 내보내기에 포함된 모든 열
- 내보내기 파일 및 해당 파일 경로의 목록. 이 목록을 프로그래밍 방식으로 읽어서 모을 파일을 식별하는 것이 좋습니다.
- 내보내기 대상 기간

모든 내보내기 데이터 파일이 S3로 전송된 후에만 Manifest.json이 전송됩니다.

내보내기 새로 고침

데이터 내보내기는 소스 데이터가 업데이트될 때마다 내보내기를 새로 고칩니다. CUR 2.0의 경우 새로 고침이 하루에 한 번 이상 발생합니다. 현재 청구 기간(파티션)은 청구 기간이 종료될 때까지 새로

고침되며, 청구 기간이 종료될 때 다음 청구 기간의 전송이 시작됩니다. 다음 청구 기간의 전송에는 해당 청구 기간의 요금 및 청구 데이터만 포함됩니다. 결제 기간이 종료된 후는 종료 후 처음 2주 이내에 이전 결제 기간에 대한 내보내기 전송을 업데이트할 AWS 수 있습니다.

내보내기 덮어쓰기 및 새로 생성

내보내기를 생성할 때 새 내보내기 파일을 생성하거나 새로 고칠 때마다 기존 내보내기 파일을 덮어쓸 수 있습니다.

새로 생성

모든 내보내기 새로 고침이 유지되므로 새 내보내기 파일을 생성하면 더 많은 S3 스토리지가 사용 됩니다. 이전 내보내기 파일을 덮어쓰면 각 청구 기간 새로 고침의 최신 버전만 유지되므로 S3 스토리지 사용량이 줄어듭니다.

'새로 생성' 모드에서는 내보내기 파일이 다음 S3 경로로 전달됩니다.

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

timestamp는 내보내기가 실행된 날짜와 시간입니다. execution-id는 실행에 할당된 고유 ID입니다.

'새로 만들기'의 경우 각 내보내기 실행과 함께 두 개의 Manifest.json 파일이 전송됩니다. 하나는 metadata/<partition>/<timestamp>-<execution-id> 디렉터리에 저장되고 다른 하나는 metadata/<partition> 디렉터리에 덮어써집니다. metadata/<partition> 디렉터리의 매니페스트는 항상 가장 최근의 새로 고침을 나타내며 해당 데이터는 가장 최근에 새로 고친 내보내기 파일의 위치를 식별하는 데 사용됩니다.

덮어쓰기.

덮어쓰기는 동일한 파티션(즉, 청구 기간)을 새로 고치는 경우에만 적용됩니다. 새 청구 기간이 시작되면 내보내기를 통해 최신 파티션이나 청구 기간에 따른 이름을 가진 새 S3 디렉터리가 생성되고 새 내보내기 파티션이 해당 디렉터리에 전달되기 시작합니다. 이전 파티션의 데이터가 업데이트되지 않는 한 이전 파티션의 내보내기는 덮어써지지 않습니다.

'덮어쓰기' 모드에서는 내보내기 파일이 다음 S3 경로로 전달됩니다.

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

이 파일 디렉터리의 내보내기 파일은 동일한 파티션(즉, 청구 기간)을 전송할 때마다 덮어써집니다.

내보내기가 충분히 커지면 내보내기 파일이 여러 개의 '청크'(별도의 gzip/csv 또는 Parquet 파일)로 전송됩니다. 한 달 동안 쿼리 변경 또는 데이터 수정으로 인해 내보내기 크기가 줄어들면 내보내기 새로 고침을 제공하는 데 필요한 청크 수가 줄어들 수 있습니다. 이 경우 데이터 내보내기는 마지막 새로 고침의 추가 청크를 빈 데이터로 덮어씁니다.

덮어쓰기의 경우 내보내기가 실행될 때마다 Manifest.json 파일 하나가 전달됩니다. 파일은 metadata/<partition> 디렉터리에 저장되며 새로 고칠 때마다 덮어써집니다.

내보내기 데이터 파일 이름 및 청크

내보내기는 한 실행의 결과를 하나의 파일(gzip/csv 또는 Parquet)로 전송하지만 내보내기의 크기가 충분히 커지면 한 실행의 결과를 여러 '청크'(별도의 gzip/csv 또는 Parquet 파일)로 전송합니다.

gzip/csv 파일 형식의 경우 내보내기의 이름은 다음과 같이 지정됩니다.

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

Parquet 형식의 경우 내보내기의 이름은 다음과 같이 지정됩니다.

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

청크 번호는 항상 5자리입니다. 청크 번호는 00001에서 시작하여 열거됩니다.

요약

새로 생성 디렉터리의 경우 내보내기 데이터 파일 이름

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
```

덮어쓰기 디렉터리의 경우 내보내기 데이터 파일 이름

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
```

새로 생성 디렉터리의 경우 매니페스트 파일 이름

'새로 생성' 모드는 Manifest.json을 두 위치로 전송합니다.

첫 번째 위치는 내보내기의 특정 실행을 나타내는 폴더(이름은 timestamp 및 execution-id)의 내부입니다. 이 매니페스트는 해당 특정 실행에 상응합니다. 파일 경로는 다음과 같습니다.

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

두 번째 위치는 모든 실행이 들어 있는 파티션 폴더의 내부입니다. 이 매니페스트는 가장 최근에 실행한 내보내기 파일과 동일합니다. 이 매니페스트를 읽고 모든 최근 내보내기 파일의 정확한 파일 경로를 확인할 수 있습니다. 파일 경로는 다음과 같습니다.

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

덮어쓰기 디렉터리의 경우 매니페스트 파일 이름

'덮어쓰기' 모드는 Manifest.json을 한 위치에 전송됩니다.

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

지정된 파티션(즉, 청구 기간)을 새로 고칠 때마다 이 디렉터리의 매니페스트가 덮어써집니다.

내보내기 세부 정보 편집

AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지를 사용하여 내보내기 세부 정보를 편집할 수 있습니다.

내보내기 세부 정보를 편집하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.

3. 내보내기 목록에서 편집하려는 내보내기 이름을 선택합니다.
4. 내보내기 세부 정보 페이지에서 편집을 선택합니다.
5. 편집 모드에서 내보내기에 대한 열 선택과 내보내기 덮어쓰기 기본 설정을 업데이트할 수 있습니다.

데이터 테이블, 추가 내보내기 콘텐츠(테이블 구성) 또는 기존 내보내기의 압축 유형 및 파일 형식은 업데이트할 수 없습니다. 이러한 세부 정보를 변경하려면 새 내보내기를 생성해야 합니다.

내보내기 태그 편집

AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지를 사용하여 내보내기 태그를 편집할 수 있습니다.

내보내기 태그를 편집하는 방법

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 내보내기 목록에서 편집하려는 내보내기 이름을 선택합니다.
4. 페이지 하단의 태그 섹션에서 태그 관리를 선택합니다.
5. 태그 관리 페이지에서 다음 작업을 수행합니다.
 - 새 태그를 추가하려면 새 태그 추가를 선택합니다. 최대 50개의 태그를 추가하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.
 - 기존 태그를 삭제하려면 제거하려는 항목 옆에 있는 제거를 선택합니다.
6. 저장을 선택합니다.

내보내기 삭제

AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지를 사용하여 내보내기를 삭제할 수 있습니다.

내보내기를 삭제하려면 다음과 같이 하세요.

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 데이터 내보내기를 선택합니다.
3. 내보내기 목록에서 삭제하려는 내보내기 이름을 선택합니다.

4. 내보내기 세부 정보 페이지에서 삭제를 선택합니다.
5. 삭제를 한 번 더 선택하여 내보내기 삭제를 확인합니다.

Note

이 절차를 수행하면 데이터 내보내기에서 내보내기가 삭제됩니다. 하지만 Amazon S3 버킷에 저장된 객체가 삭제되지는 않습니다.

Cost and Usage Dashboard의 경우 위 절차를 수행하면 데이터 내보내기에서 Cost and Usage Dashboard가 삭제됩니다. 하지만 S3 버킷에 저장된 객체, QuickSight 대시보드 및 추가 QuickSight 리소스는 삭제되지 않습니다. QuickSight에서 Cost and Usage Dashboard를 삭제하려면 [Amazon QuickSight 대시보드 삭제](#)를 참조하세요.

Amazon QuickSight 대시보드를 삭제하면 대시보드가 속해 있던 계정 및 모든 폴더에서 해당 대시보드가 영구적으로 제거됩니다. 삭제된 대시보드에는 더 이상 액세스할 수 없습니다. 소유한 또는 공동 소유한 대시보드만 삭제할 수 있습니다.

AWS 조직에서 데이터 내보내기 사용

Data Exports는 AWS Organizations와 함께 작동하여 관리 계정이 조직의 모든 계정에 대한 데이터로 내보내기를 생성할 수 있습니다. 멤버 계정도 데이터 내보내기를 생성할 수 있지만, 내보내기에는 해당 특정 멤버 계정의 결제 및 비용 관리 데이터만 포함할 수 있습니다. 관리 계정이 모든 멤버 계정에 대한 데이터를 수신하는지 여부를 제어하는 설정은 데이터 내보내기 테이블에 따라 다릅니다. 각 테이블에 멤버 계정 데이터를 포함할지 여부를 결정하는 방법에 대한 자세한 내용은 다음 섹션을 참조하세요.

- [비용 및 사용량 보고서 2.0 \(CUR 2.0\)](#)
- [비용 최적화 권장 사항\(Cost Optimization Hub에서\)](#)
- [AWS 열이 있는 FOCUS 1.0](#)
- [Cost and Usage Dashboard](#)
- [탄소 배출량](#)

내보내기 생성 기능을 허용하거나 제한하는 IAM 정책은 관리 계정과 멤버 계정 모두에서 동일합니다.

AWS Organizations 관리 계정의 관리자이고 멤버 계정이 내보내기를 생성하지 않도록 하려면 멤버 계정이 내보내기를 생성하지 못하도록 하는 서비스 제어 정책(SCP)을 적용할 수 있습니다. SCP는 멤버 계정이 새 내보내기를 만들지 못하도록 방지하지만 이전에 만든 내보내기는 삭제하지 않습니다.

Note

SCP는 멤버 계정에만 적용할 수 있습니다. 관리 계정이 보고서를 생성하지 못하게 하려면 관리 계정의 사용자 역할에 연결된 IAM 정책을 수정합니다.

데이터 내보내기 테이블 사전

데이터 내보내기는 내보내기를 생성할 때 쿼리할 수 있는 여러 테이블을 제공합니다.

다음 주제에서는 사용 가능한 각 테이블의 스키마를 각 열에 대한 정의와 함께 설명합니다.

주제

- [비용 및 사용량 보고서\(CUR\) 2.0](#)
- [비용 최적화 권장 사항\(Cost Optimization Hub에서\)](#)
- [AWS 열이 있는 FOCUS 1.2](#)
- [AWS 열이 있는 FOCUS 1.0](#)
- [Cost and Usage Dashboard](#)
- [탄소 배출량](#)

비용 및 사용량 보고서(CUR) 2.0

CUR 2.0 테이블은 몇 가지 개선 사항을 제외하고 비용 및 사용량 보고서(CUR)와 동일한 정보를 제공합니다.

비용 및 사용량 보고서 2.0은 비용 및 사용량 보고서에 비해 다음과 같은 개선 사항을 제공합니다.

- 일관된 스키마: CUR 2.0에는 고정된 열 세트가 포함된 반면, CUR에 포함된 열은 AWS 서비스 사용, 비용 범주 및 리소스 태그에 따라 매월 다를 수 있습니다.
- 중첩 데이터: CUR 2.0은 CUR의 열의 키값 쌍이 있는 개별 열로 특정 열을 축소하여 데이터 희소성을 줄입니다. 필요에 따라 데이터 내보내기에서 원래 CUR 스키마 및 데이터와 일치하도록 중첩된 키를 별도의 열로 쿼리할 수 있습니다.
- 추가 열: CUR 2.0에는 bill_payer_account_name, line_item_usage_account_name, capacity_reservation_capacity_reservation_arn, capacity_reservation_capacity_reservation_status 및 capacity_reservation_capacity_reservation_type과 같은 추가 열이 포함되어 있습니다.

CUR 2.0의 SQL 테이블 이름은 COST_AND_USAGE_REPORT입니다.

테이블 구성

테이블 구성은 데이터 내보내기에서 테이블을 쿼리하기 전에 사용자가 테이블의 데이터나 스키마를 변경하도록 설정할 수 있는 사용자 제어 속성입니다. 테이블 구성은 JSON 문으로 저장되며 AWS SDK/CLI의 사용자 입력 또는 콘솔의 사용자 선택을 통해 지정됩니다.

CUR 2.0에는 다음과 같은 테이블 구성이 있습니다.

구성 이름	설명	유효값
TIME_GRANULARITY	<p>이 구성은 CUR 2.0 테이블의 비용 및 사용량 품목이 다른 시간 세부 수준을 갖도록 변경합니다.</p> <p>예를 들어, 'HOURLY'를 선택하면 모든 품목이 한 시간의 사용량을 나타냅니다.</p>	HOURLY, DAILY, MONTHLY
INCLUDE_RESOURCES	<p>이 구성은 CUR 2.0 테이블의 비용 및 사용량 품목을 리소스 수준으로 세분화하여 변경하고 'line_item_resource_id' 열을 테이블 스키마에 추가합니다.</p> <p>이 구성을 활성화하면 CUR 2.0 테이블에 해당 서비스의 총 사용량을 합산하여 표시하는 대신 해당 서비스의 사용량이 발생한 각 리소스에 대한 품목이 표시됩니다.</p> <p>이 구성을 활성화하면 파일 크기와 내보내기의 행 수가 크게 늘어날 수 있습니다.</p>	TRUE, FALSE
INCLUDE_SPLIT_COST_ALLOCATION_DATA		TRUE, FALSE

구성 이름	설명	유효값
	<p>이 구성은 CUR 2.0 테이블에 분할 비용 할당 데이터 및 열 (split_line_item_*)을 추가합니다. 이 데이터는 특정 AWS 리소스의 사용량을 여러 사업부 또는 팀에 할당할 수 있는 방법을 나타냅니다.</p> <p>이 구성을 활성화하면 EC2 인스턴스에서 실행되는 여러 컨테이너에 EC2 인스턴스를 할당하는 방법을 보여주는 행과 열을 추가할 수 있습니다. 자세한 내용은 분할 비용 할당 데이터 이해를 참조하세요.</p>	
INCLUDE_CAPACITY_RESERVATION_DATA	<p>참고:이 구성은 2025년 11월 1일부터 새 열에 데이터만 추가합니다.</p> <p>이 구성을 활성화하면 인스턴스 사용량이 여러 용량 예약으로 분할되거나 한 시간 내에 용량 예약에서 부분적으로 사용될 때 리소스 수준 세부 수준을 갖도록 CUR 2.0 테이블의 비용 및 사용량 항목이 변경됩니다. 또한 테이블 스키마에 3개의 새 열이 추가되어 용량 예약에서 EC2 인스턴스가 시작되는 방식을 보여줍니다.</p>	TRUE, FALSE

구성 이름	설명	유효값
INCLUDE_IAM_PRINCIPAL_DATA	<p>참고:이 구성은 2026년 4월 8일부터 새 열에 데이터만 추가합니다.</p> <p>이 구성을 활성화하면 CUR 2.0 테이블에 Amazon Bedrock 모델 추론 비용에 대한 호출자 자격 증명(IAM 보안 주체) 할당 데이터가 포함됩니다. 그러면 Bedrock 추론 요청을 수행하는 호출자의 AWS IAM 보안 주체 ARN이 포함된 line_item_iam_principal 열이 테이블 스키마에 추가됩니다. IAM 보안 주체 태그가 비용 할당 태그로 활성화되면 해당 태그는 iamPrincipal/ 접두사 (예: iamPrincipal/department, iamPrincipal/cost-center)가 있는 태그 열에도 표시됩니다. 이 구성을 활성화하면 CUR 행 수와 파일 크기가 증가할 수 있습니다.</p>	TRUE, FALSE

구성 이름	설명	유효값
INCLUDE_MANUAL_DISCOUNT_COMPATIBILITY	<p>참고: 이 구성은 할인이 자동으로 계산되는 할인 자동화 프로그램에 가입한 AWS 고객에게만 적용됩니다.</p> <p>이 구성은 CUR 2.0 테이블의 할인을 CUR에 '수동으로' 추가했을 때와 같이 일반적으로 별도의 품목으로 나타나도록 변경하고 스키마에서 두 개의 열('discount' 및 'total_discount')을 제거합니다.</p>	TRUE, FALSE

AWS 조직 지원

CUR 2.0 테이블은 AWS Organizations의 통합 결제 기능에서 설정한 설정을 상속합니다. 통합 결제가 활성화되면 관리 계정과 멤버 계정에 대한 다양한 동작이 존재합니다. 관리 계정을 사용량 경우 CUR 2.0 테이블에는 관리 계정 및 조직의 모든 멤버 계정에 대한 비용 및 사용 데이터가 포함됩니다. 멤버 계정을 사용할 경우 CUR 2.0 테이블에는 해당 멤버 계정의 비용 및 사용 데이터만 포함됩니다.

조직에 가입한 후 멤버 계정은 해당 계정이 조직의 멤버로 있었던 기간 동안의 데이터만 내보낼 수 있습니다. 매월 15일에 조직 A를 떠나 조직 B에 가입하는 멤버 계정을 예로 들어 보겠습니다. 그런 다음 멤버 계정이 보고서를 생성합니다. 멤버 계정이 조직 B에 가입한 후 내보내기를 만들었기 때문에 해당 월의 멤버 계정 CUR 2.0 내보내기에는 해당 계정이 조직 B의 멤버로 활동한 기간에 대한 비용 및 사용 데이터만 포함됩니다.

멤버 계정이 새 조직에 가입하면 멤버 계정의 비용 및 사용 데이터가 새 조직의 내보내기에 기록됩니다. 이는 멤버 계정으로 전환되어 새 조직에 가입하는 관리 계정의 경우와 동일한 결과입니다.

멤버 계정이 조직을 탈퇴하거나 독립 실행형 계정으로 전환되는 경우, 멤버 계정은 이전 내보내기가 저장되는 Amazon S3 버킷에 대한 권한이 있는 한 이전 내보내기에 계속 액세스할 수 있습니다.

자세한 내용은 [결제 사용 설명서의 AWS 조직에 대한 통합 결제](#)를 참조하세요. AWS

CUR 2.0 열 그룹

CUR 2.0 테이블에는 다음과 같이 그룹화된 125개의 열이 있습니다.

- 청구서: 청구 기간의 청구서 관련 데이터입니다.
- 비용 범주: 품목에 적용되는 비용 범주에 대한 데이터입니다.
- 용량 예약: 해당 품목에 적용되는 용량 예약에 대한 데이터입니다.
- 할인: 현재 받고 있는 모든 할인에 대한 데이터입니다.
- ID: 품목을 식별하기 위한 데이터입니다.
- 품목: 비용, 사용량, 사용 유형, 가격 요율, 제품 이름 등에 대한 데이터입니다.
- 요금: 품목별 요금에 대한 데이터입니다.
- 제품: 품목에서 요금이 청구되는 제품에 대한 데이터입니다.
- 예약: 품목에 적용되는 예약에 대한 데이터입니다.
- 리소스 태그: 품목에 적용되는 리소스 태그에 대한 데이터입니다.
- 절감형 플랜: 품목에 적용되는 절감형 플랜에 대한 데이터입니다.
- 분할 품목: 다른 품목에 대한 분할 비용 할당에 관한 데이터입니다.
- 용량 예약: 품목에 적용되는 용량 예약에 대한 데이터입니다.
- 태그: 사용자, 계정, 비용 범주 및 리소스 태그에 대한 데이터입니다.

청구서 열

청구서 열에는 청구 기간의 청구서 관련 데이터가 포함됩니다.

열 이름	설명	데이터 유형
bill_bill_type	<p>이 보고서가 다루는 결제 유형입니다. 3가지 청구 유형이 있습니다.</p> <ul style="list-style-type: none"> •결제일: 한 달 동안 사용한 서비스의 품목입니다. • 	string

열 이름	설명	데이터 유형
	<p>구매: 선결제 서비스 요금의 품목입니다.</p> <ul style="list-style-type: none"> • 환급: 환급된 품목입니다. 	
bill_billing_entity	인보이스 또는 트랜잭션이 AWS Marketplace용인지 아니면 다른 AWS 서비스 구매용인지 식별하는 데 도움이 됩니다.	string
bill_billing_period_end_date	이 보고서의 청구 기간 종료일 (UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ 입니다.	timestamp
bill_billing_period_start_date	이 보고서의 청구 기간 시작일 (UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ 입니다.	timestamp
bill_invoice_id	특정 항목과 연결된 ID입니다. 보고서가 완성될 때까지 InvoiceId 는 비어 있습니다.	string
bill_invoicing_entity	인보이스를 발행하는 AWS 엔터티입니다.	string
bill_payer_account_id	지급 계정의 계정 ID입니다. AWS Organizations에 속한 조직의 경우 이 항목은 관리 계정의 계정 ID입니다.	string

열 이름	설명	데이터 유형
bill_payer_account_name	지급 계정의 계정 이름입니다. AWS Organizations에 속한 조직의 경우 이 항목은 관리 계정의 이름입니다.	string

비용 범주 열

비용 범주 열에는 품목에 적용되는 비용 범주에 대한 데이터가 포함됩니다. 리소스 태그도 태그 열에 포함되므로 태그 열을 선택한 경우 이 열을 선택할 필요가 없습니다.

열 이름	설명	데이터 유형
cost_category	<p>주어진 품목에 대한 비용 범주와 해당 값의 키-값 쌍이 포함된 맵 열입니다. 이러한 키와 값은 비용 범주 기능에서 생성한 분류 규칙에 따라 채워집니다.</p> <p>비용 범주 키는 특정 품목에 적용되는 값이 있는 경우에만 맵 열에 표시됩니다.</p> <p>점 연산자를 사용하여 이 열의 키를 개별 열로 쿼리할 수 있습니다. 자세한 내용은 데이터 쿼리를 참조하세요.</p>	map <string, string>

할인 열

할인 열에는 현재 받고 있는 모든 할인에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
discount	<p>테이블 구성: 제거 기준: INCLUDE MANUAL DISCOUNT COMPATIBILITY</p> <p>이 품목에 적용되는 특정 할인의 키-값 쌍을 포함하는 '구조체' 열입니다. 키는 할인 유형에 해당하고 값은 할인 값 또는 기타 정보에 해당합니다. 이 열의 값은 키에 따라 데이터 유형이 '숫자' 또는 '문자열'입니다.</p> <p>점 연산자를 사용하여 이 열의 키를 개별 열로 쿼리할 수 있습니다. 자세한 내용은 데이터 쿼리를 참조하세요.</p> <p>'수동 할인 호환성'이 활성화된 경우에는 이 열을 사용할 수 없습니다. 활성화된 경우 할인은 이 열이 아닌 별도의 품목으로 채워집니다.</p>	map <string, double>
discount_bundled_discount	<p>번들 할인은 항목에 적용됩니다. 번들 할인은 다른 서비스 또는 기능의 사용에 따라 서비스 또는 기능을 무료 또는 할인된 가격으로 제공하는 사용량 기반 할인입니다.</p> <p>2025년 8월부터는 '소유자 우선 접근 방식'을 통해 번들 할인이 적용되며, 이 방식에서는 소</p>	double

열 이름	설명	데이터 유형
	<p>스 사용량을 생성하는 계정에 먼저 할인이 적용됩니다. 소스 계정 내에서 할인은 다음 순서에 따라 적용됩니다.</p> <ul style="list-style-type: none"> • 단위당 최고 가격 • 이전 사용 타임스탬프 • 다른 요인이 동일할 때 더 많은 사용량 <p>잔여 할인은 통합 결제 패밀리 (CBF) 내 다른 계정에 다음 순서에 따라 배분됩니다.</p> <ul style="list-style-type: none"> • 단위당 최고 가격 • 계정 ID 오름차순 • 이전 사용 타임스탬프 • 더 많은 사용량 <p>번들 할인의 예는 다음과 같습니다.</p> <ul style="list-style-type: none"> • Shield Advanced AWS 를 사용하는 경우 AWS WAF 비용을 별도로 지불할 필요가 없습니다. AWS WAF 사용량은 AWS Shield Advanced 와 함께 번들로 제공됩니다. AWS Shield Advanced에 대한 자세한 내용은 Amazon 	

열 이름	설명	데이터 유형
	<p>CloudFront 요금을 참조하세요.</p> <ul style="list-style-type: none"> AWS Network Firewall을 사용하여 NAT 게이트웨이를 생성하는 경우 방화벽의 GB 및 사용 시간당 처리와 함께 표준 NAT 게이트웨이 처리 및 시간당 사용 요금이 one-to-one로 면제됩니다. 자세한 내용은 AWS Network Firewall 요금을 참조하세요. Amazon Interactive Video Service(IVS) 채팅을 사용하면 1시간의 비디오 입력을 전송할 때마다 추가 비용 없이 2,700개의 전송 메시지와 270,000개의 전달된 메시지를 받을 수 있습니다. 자세한 내용은 Amazon Interactive Video Service 요금을 참조하세요. 	

열 이름	설명	데이터 유형
discount_total_discount	<p>테이블 구성: 제거 기준: INCLUDE MANUAL DISCOUNT COMPATIBILITY</p> <p>해당 항목에 대한 모든 할인 열의 합계입니다.</p> <p>'수동 할인 호환성'이 활성화된 경우에는 이 열을 사용할 수 없습니다. 활성화된 경우 할인은 이 열이 아닌 별도의 품목으로 채워집니다.</p>	double

ID 열

ID 열에는 품목을 식별하는 데이터가 포함됩니다.

열 이름	설명	데이터 유형
identity_line_item_id	<p>이 필드는 각 품목에 대해 생성되며 지정된 파티션마다 고유합니다. 이렇게 AWS 해도 필드가 CUR의 전체 전송(즉, 업데이트의 모든 파티션)에서 고유하다는 보장은 없습니다. 항목 ID는 각기 다른 비용 및 사용 보고서 사이에서 일관적이지 않으며, 각기 다른 보고서에 걸쳐 동일한 항목을 식별하는 데 사용할 수 없습니다.</p>	string
identity_time_interval		string

열 이름	설명	데이터 유형
	이 항목에 적용되는 시간 간격은 다음 형식을 갖습니다. YYYY-MM-DDTHH:mm:ssZ/YYYY-MM-DDTHH:mm:ssZ . 시간 간격은 UTC로 표시되고, 보고서의 세부 수준에 따라 일별 또는 시간별이 될 수 있습니다.	

품목 열

품목 열에는 비용, 사용량, 사용 유형, 가격 효율, 제품 이름 등에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
line_item_usage_account_name	이 품목을 사용한 계정의 이름입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다. 이 필드를 사용하여 계정별 비용과 사용량을 추적할 수 있습니다.	string
line_item_availability_zone	이 항목을 호스팅하는 가용 영역입니다. 예: us-east-1a 또는 us-east-1b .	string
line_item_blended_cost	BlendedRate 에 UsageAmount 를 곱한 값과 같습니다. LineItemType이 할인인 항목의 경우 BlendedCost는 비어 있습니다. 할인은 멤버 계정의 일반 비용만 사용하여 계산되며 멤	double

열 이름	설명	데이터 유형
	<p>버 계정 및 SKU별로 집계됩니다. 따라서 BlendedCost는 할인에 사용할 수 없습니다.</p>	
line_item_blended_rate	<p>BlendedRate 는 조직 전체의 각 SKU에 대해 발생하는 평균 비용을 나타냅니다.</p> <p>예를 들어 Amazon S3 혼합 요금은 월별로 저장된 데이터의 양으로 분할된 스토리지의 총 비용입니다. RI가 포함된 계정의 경우 혼합 요금은 RI 및 온디맨드 인스턴스의 평균 비용으로 계산됩니다.</p> <p>혼합 요금은 관리 계정 수준에서 계산되며 각 멤버 계정에 비용을 할당하는 데 사용됩니다. 자세한 내용은 AWS 사용 설명서의 혼합 요금 및 요금을 참조하세요.</p>	string
line_item_currency_code	<p>이 항목에 표시된 통화입니다. 모든 AWS 고객에게는 기본적으로 미국 달러로 요금이 청구됩니다. 청구 통화를 변경하려면 AWS Billing 사용 설명서의 청구서 결제에 사용하는 통화 변경을 참조하세요.</p>	string

열 이름	설명	데이터 유형
line_item_iam_principal	Amazon Bedrock 모델 추론을 수행한 보안 주체의 IAM ARN입니다. 이 열은 CUR 2.0 데이터 내보내기에서 IAM 보안 주체 데이터를 활성화할 때 채워집니다. 현재 Amazon Bedrock에서만 지원됩니다.	string
line_item_legal_entity	<p>특정 제품 또는 서비스의 등록 판매자. 대부분의 경우 인보이스 발행 주체와 법인은 동일합니다. 타사 AWS Marketplace 트랜잭션의 경우 값이 다를 수 있습니다. 가능한 값은 다음과 같습니다.</p> <ul style="list-style-type: none"> Amazon Web Services, Inc.: AWS 서비스를 판매하는 기업입니다. Amazon Web Services India Private Limited: 인도에서 AWS 서비스의 리셀러 역할을 하는 인도 현지 법인입니다. 	string

열 이름	설명	데이터 유형
line_item_line_item_description	<p>항목 유형의 설명입니다. 예를 들어 사용량 품목의 설명은 지정된 기간 동안 발생한 사용량 유형을 요약합니다.</p> <p>유연한 크기의 RI의 경우 설명은 그 혜택이 적용되는 RI와 일치합니다. 예를 들어 품목이 t2.micro에 해당하고 t2.small RI가 사용량에 적용된 경우 line_item_line_item_description에 t2.small이 표시됩니다.</p> <p>RI 할인이 포함된 사용량 항목의 설명에는 항목에 포함되는 요금제를 포함합니다.</p>	string

열 이름	설명	데이터 유형
line_item_line_item_type	<p>이 항목에 적용되는 요금 유형입니다. 가능한 데이터 유형은 다음과 같습니다.</p> <ul style="list-style-type: none"> • BundledDiscount: 다른 서비스 또는 기능의 사용에 따라 서비스 또는 기능을 무료 또는 할인된 가격으로 사용할 수 있는 사용량 기반 할인입니다. • 크레딧: 청구서에 AWS 적용된 모든 크레딧입니다. 가 청구서를 완료한 후 해당 월의 계정에 크레딧을 AWS 적용하는 경우, 완료된 후 details. AWS might 업데이트 보고서는 설명 열을 참조하세요. • 할인: 사용량에 AWS 적용된 모든 할인입니다. 이 특정 항목의 이름은 다를 수 있으며 할인을 기준으로 구문 분석해야 합니다. 자세한 내용은 lineItem/LinItemDescription 열을 참조하세요. • DiscountedUsage: 예약형 인스턴스(RI) 혜택을 보유한 모든 인스턴스의 요금입니다. • Fee: 구독에 대해 지불하는 모든 선결제 연간 요금입니다. 	string

열 이름	설명	데이터 유형
	<p>다. 예를 들어 전체 선결제 RI 또는 부분 선결제 RI에 대해 지불하는 선결제 요금입니다.</p> <ul style="list-style-type: none"> • 환급: AWS 에서 환급한 음 수 요금입니다. 가 청구서를 완료한 후 해당 월의 계정에 환급을 AWS 적용하는 경우, 완료 후 설명 열에서 details. AWS might 업데이트 보고서를 검토합니다. • RIFee: 구독에 대해 매달 발생하는 요금입니다. 예를 들 면, 부분 선결제 RI, 선결제 없음 RI 및 전체 선결제에 대 해 매달 지불하는 기본 요금 입니다. 모든 선결제 예약에 대해 RIFee가 0 USD일 수 있지만, 이 행은 reservation/ AmortizedUpfrontFeeForBi llingPeriod 및 reservation/ ReservationARN과 같은 다 른 열을 제공하기 위해 해당 예약 유형에 대해 여전히 채 워집니다. • Tax: AWS 가 청구서에 적용 하는 모든 세금입니다. 예: VAT 또는 미국 판매세. • Usage: 온디맨드 인스턴스 요금이 부과되는 모든 사용 량입니다. 	

열 이름	설명	데이터 유형
	<ul style="list-style-type: none"> • SavingsPlanUpfrontFee: 전액 선결제 또는 부분 선결제 절감형 플랜 구매 시 발생하는 일회성 선결제 요금입니다. • SavingsPlanRecurringFee: 선결제 없음 또는 부분 선결제 절감형 플랜에 해당하는 모든 시간당 기본 요금입니다. 절감형 플랜 기본 요금은 선결제 없음 또는 부분 선결제 절감형 플랜을 구매한 날에 청구서에 처음으로 가산됩니다. 최초 구매 후에는 이후 각 결제 기간의 첫날에 반복 요금을 AWS 추가합니다. • SavingsPlanCoveredUsage: 절감형 플랜이 적용되는 모든 온디멘드 비용입니다. 절감형 플랜 적용 사용량 항목은 상응하는 절감형 플랜 부정 항목에 의해 상쇄됩니다. • SavingsPlanNegation - 상응하는 절감형 플랜 적용 사용량 항목과 관련된 절감형 플랜 혜택을 통해 상쇄된 비용입니다. 	

열 이름	설명	데이터 유형
line_item_net_unblended_cost	항목에 대해 지불하는 실제 할인 후 비용입니다. 이 열은 해당 청구 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
line_item_net_unblended_rate	항목에 대해 지불하는 실제 할인 후 요금입니다. 이 열은 해당 청구 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	string
line_item_normalization_factor	인스턴스가 테넌시를 공유한 경우는 인스턴스 패밀리 및 AWS 리전의 모든 인스턴스 크기에 모든 리전 Linux 또는 Unix Amazon EC2 및 Amazon RDS RI 할인을 적용할 AWS 수 있습니다. 또한 조직의 멤버 계정에 대해 RI 할인을 적용합니다. 모든 신규 및 기존 Amazon EC2 및 Amazon RDS 유연한 크기 RI는 인스턴스 크기를 기반으로 하는 정규화 인자에 따라 크기 조정됩니다.	double
line_item_normalized_usage_amount	유연한 크기의 RI에 대해 정규화 유닛에서 발생한 사용량입니다. NormalizeUsageAmount는 UsageAmount에 NormalizationFactor를 곱한 값과 같습니다.	double

열 이름	설명	데이터 유형
line_item_operation	이 행 항목에서 다루는 특정 AWS 작업입니다. 이 항목의 구체적인 사용량에 대해 설명합니다. 예를 들어 RunInstances 값은 Amazon EC2 인스턴스 작업을 나타냅니다.	string
line_item_product_code	측정된 제품의 코드입니다. 예를 들어 Amazon EC2는 Amazon Elastic Compute Cloud의 제품 코드입니다.	string
line_item_resource_id	테이블 구성: INCLUDE RESOURCES에 의해 추가됨 (선택 사항) 보고서에 개별 리소스 ID를 포함하기로 선택했다면, 이 열은 프로비저닝한 리소스 ID를 포함합니다. 예를 들어 Amazon S3 스토리지 버킷, Amazon EC2 컴퓨팅 인스턴스 또는 Amazon RDS 데이터베이스 각각에 리소스 ID가 있습니다. 데이터 전송 및 API 요청과 같이 인스턴스화된 호스트와 연결되지 않은 사용량 유형, 그리고 할인, 크레딧 및 세금과 같은 항목 유형에 대해서는 이 필드가 빈 칸입니다.	string
line_item_tax_type	이 항목에 AWS 적용된 세금 유형입니다.	string

열 이름	설명	데이터 유형
line_item_unblended_cost	UnblendedCost 는 UnblendedRate 에 UsageAmount 를 곱한 값입니다.	double
line_item_unblended_rate	<p>AWS Organizations를 사용하는 계정에 대한 통합 결제에서 일반 요금은 개별 계정의 서비스 사용과 관련된 요금입니다.</p> <p>RI 할인이 적용된 Amazon EC2 및 Amazon RDS 품목의 경우 UnblendedRate 는 0입니다. RI 할인이 있는 항목의 LineItemType 는 DiscountedUsage 입니다.</p>	string
line_item_usage_account_id	이 항목에서 사용된 계정의 계정 ID입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다. 이 필드를 사용하여 계정별 비용과 사용량을 추적할 수 있습니다.	string
line_item_usage_amount	<p>지정된 기간 동안 발생한 사용량입니다. 크기가 유연한 모든 예약형 인스턴스에는 reservation/TotalReservedUnits 열을 대신 사용하세요.</p> <p>특정 구독 요금의 UsageAmount는 0입니다.</p>	double

열 이름	설명	데이터 유형
line_item_usage_end_date	해당 항목의 종료 날짜와 시간(UTC)입니다. 배타적입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ 입니다.	timestamp
line_item_usage_start_date	이 항목의 시작 날짜와 시간(UTC)입니다. 포괄적입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ 입니다.	timestamp
line_item_usage_type	이 항목의 사용량 세부 정보입니다. 예를 들어 USW2-BoxUsage:m2.2xlarge 는 미국 서부(오레곤) 리전에 있는 M2 고용량 메모리 더블 엑스트라 라지 인스턴스를 설명합니다.	string
line_item_user_identifier	작업 인력 사용자의 Identity Access Management(IAM) Identity Center 식별자입니다. 월별 정액 구독 및 온디맨드 요금은 이 식별자로 식별되는 사용자에게 대해 계산됩니다.	string

요금 열

요금 열에는 품목별 요금에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
pricing_currency	요금 데이터가 표시되는 통화입니다.	string
pricing_lease_contract_length		string

열 이름	설명	데이터 유형
	RI가 예약되어 있는 기간입니다.	
pricing_offering_class	예약형 인스턴스에서 제공하는 클래스입니다.	string
pricing_public_on_demand_cost	퍼블릭 온디맨드 인스턴스 요금 기반의 항목의 총 비용. 여러 온디맨드 퍼블릭 요금이 있는 SKU를 보유하고 있는 경우 가장 높은 티어에 상응하는 요금이 표시됩니다. 예를 들어 프리 티어 또는 계층화된 요금을 제공하는 서비스가 있습니다.	double
pricing_public_on_demand_rate	특정 사용 항목에 대한 이 결제 기간 내의 퍼블릭 온디맨드 인스턴스 요금. 여러 온디맨드 퍼블릭 요금이 있는 SKU를 보유하고 있는 경우 가장 높은 티어에 상응하는 요금이 표시됩니다. 예를 들어 프리 티어 또는 계층화된 요금을 제공하는 서비스가 있습니다.	string
pricing_purchase_option	어떻게 이 항목에 지불할 결정을 했습니까. 유효한 값은 All Upfront, Partial Upfront, No Upfront입니다.	string

열 이름	설명	데이터 유형
pricing_rate_code	제품/제안/요금 티어 조합의 고유 코드. 제품 및 조건 조합은 요금 차원이 여러 개 있을 수 있습니다(예: 프리 티어, 저사용 티어, 고사용 티어 등).	string
pricing_rate_id	항목에 대한 요금의 ID입니다.	string
pricing_term	AWS 사용량이 예약인지 온디맨드인지 여부입니다.	string
pricing_unit	사용 비용을 계산하는 데 AWS 사용되는 요금 단위입니다. 예를 들어 Amazon EC2 인스턴스 사용량의 요금 단위는 시간입니다.	string

제품 열

제품 열에는 품목에서 요금이 청구되는 제품에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
product	<p>특정 품목에 대한 여러 제품 속성과 값의 키-값 쌍이 포함된 맵 열입니다.</p> <p>제품 속성은 특정 품목에 적용되는 값이 있는 경우에만 맵 열에 나타납니다.</p>	map <string, string>

열 이름	설명	데이터 유형
	<p>Note</p> <p>CUR에는 표시되었지만 레거시 CUR 2.0 정적 스키마에 속하지 않는 모든 제품 열이 이 맵 열에 표시됩니다.</p> <p>점 연산자를 사용하여 이 열의 키를 개별 열로 쿼리할 수 있습니다. 자세한 내용은 데이터 쿼리를 참조하세요.</p>	
product_comment	제품에 대한 설명입니다.	string
product_fee_code	요금을 나타내는 코드입니다.	string
product_fee_description	제품 요금에 대한 설명입니다.	string
product_from_location	사용량이 발생한 위치를 설명합니다.	string
product_from_location_type	사용량이 발생한 위치 유형을 설명합니다.	string
product_from_region_code	AWS 서비스의 소스 리전 코드를 설명합니다.	string
product_instanceSKU	제품 인스턴스의 SKU입니다.	string

열 이름	설명	데이터 유형
product_instance_family	Amazon EC2 인스턴스 패밀리를 설명합니다. Amazon EC2는 10개의 서로 다른 인스턴스 유형에 대해 다양한 옵션을 제공합니다. 유형마다 하나 이상의 크기 옵션이 제공되며, 다양한 애플리케이션 유형에 맞게 최적화된 개별 인스턴스 패밀리로 구성되어 있습니다.	string
product_instance_type	인스턴스의 CPU, 네트워킹 및 스토리지 용량을 정의하는 인스턴스 유형, 크기 및 패밀리를 설명합니다.	string
product_location	Amazon S3 버킷이 있는 리전을 설명합니다.	string
product_location_type	작업의 엔드포인트를 설명합니다.	string
product_operation	이 행 항목이 다루는 특정 AWS 작업을 설명합니다.	string
product_pricing_unit	AWS 서비스에 대한 가장 작은 결제 단위입니다. 예를 들어 API 직접 호출당 0.01c입니다.	string
product_product_family	제품 유형의 범주.	string

열 이름	설명	데이터 유형
product_region_code	리전은 데이터 센터가 클러스터되어 있는 전 세계의 물리적 위치입니다. AWS 는 각 논리 데이터 센터 그룹을 가용 영역 (AZ)이라고 부릅니다. 각 AWS 리전은 지리적 영역 내에서 물리적으로 분리된 여러 개의 격리된 AZs로 구성됩니다. 리전 코드 속성은 AWS 리전과 이름이 동일하며 AWS 서비스를 사용할 수 있는 위치를 지정합니다.	string
product_sku	제품의 고유 코드. SKU는 ProductCode , UsageType 및 Operation 을 결합하여 생성됩니다. 유연한 크기의 RI 의 경우 SKU가 사용되었던 인스턴스를 사용합니다. 예를 들어 t2.micro 인스턴스를 사용하고 사용량에 t2.small RI 할인을 AWS 적용한 경우를 사용하여 품목 SKU가 생성됩니다. t2.micro.	string
product_servicecode	이는 고객에 대한 특정 AWS 서비스를 고유한 짧은 약어로 식별합니다.	string
product_to_location_type	서비스 사용의 대상 위치를 설명합니다.	string

열 이름	설명	데이터 유형
product_to_location	위치 사용 대상을 설명합니다.	string
product_to_region_code	AWS 서비스의 소스 리전 코드를 설명합니다.	string
product_usagetype	품목의 사용량 세부 정보를 설명합니다.	string

예약 열

예약 열에는 품목에 적용되는 예약에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
reservation_amortized_upfront_cost_for_usage	사용 시간 동안 상각되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다. 값은 다음과 같습니다. $RIAmortizedUpfrontFeeForBillingPeriod * \frac{\text{The normalized usage amount for DiscountedUsage line items}}{\text{The normalized usage amount for the RIFee}}$. 선결제 없음 RI의 경우 선결제 금액이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.	double

열 이름	설명	데이터 유형
reservation_amortized_upfront_fee_for_billing_period	이 예약의 선결제 요금이 이 결제 기간 동안 얼마나 부과되는지를 알려줍니다. 이 달 동안 분할 상환되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다. 선결제 없음 RI의 경우 선결제 요금이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.	double
reservation_availability_zone	이 항목과 연결된 리소스의 가용 영역.	string
reservation_effective_cost	유효한 시간당 요금으로 평균을 낸 RI의 선결제 및 시간당 요금의 합입니다. Effective Cost 는 amortized UpfrontCostForUsage 를 가져온 다음 recurring FeeForUsage 에 더하여 계산됩니다.	double
reservation_end_time	연결된 RI 리스 기간의 마지막 날짜입니다.	string

열 이름	설명	데이터 유형
reservation_modification_status	<p>RI 리스가 수정되었는지 아니면 변경된 사항이 없는지 보여줍니다.</p> <p>원본: 구입한 RI가 변경된 적이 없습니다.</p> <p>시스템: 구입한 RI가 콘솔 또는 API를 사용하여 변경되었습니다.</p> <p>수동: 구매한 RI가 AWS 지원 지원을 사용하여 수정되었습니다.</p> <p>ManualWithData: 구매한 RI는 AWS Support 지원을 사용하여 수정되었으며 AWS RI에 대한 추정치를 계산했습니다.</p>	string
reservation_net_amortized_upfront_cost_for_usage	<p>사용 시간 동안 분할 상환되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다 (해당하는 경우). 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.</p>	double
reservation_net_amortized_upfront_fee_for_billing_period	<p>청구 기간 동안의 예약 선결제 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.</p>	double

열 이름	설명	데이터 유형
reservation_net_effective_cost	유효한 시간당 요금으로 평균을 낸 RI의 선결제 및 시간당 요금의 합입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
reservation_net_recurring_fee_for_usage	기본 사용 요금의 할인 후 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
reservation_net_unused_amortized_upfront_fee_for_billing_period	청구 기간의 미사용 분할 상환 선결제 요금입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
reservation_net_unused_recurring_fee	할인 후 부분 선결제 및 선결제 없음 RI에 대해 미사용 예약 시간과 연결된 기본 요금입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
reservation_net_upfront_value	할인이 적용된 RI의 선결제 금액입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double

열 이름	설명	데이터 유형
reservation_normalized_units_per_reservation	예약 구독의 각 인스턴스에 대한 정규화 단위의 수입입니다.	string
reservation_number_of_reservations	이 구독권이 적용되는 예약의 수입입니다. 예를 들어 RI 구독권 하나에 연관 RI 예약 4개가 포함될 수 있습니다.	string
reservation_recurring_fee_for_usage	부분 선결제 RI 및 선결제 없음 RI의 경우, 사용 시간 동안 상각되는 기본 요금입니다. 값은 다음과 같습니다. $\text{The unblended cost of the RIFee} = \frac{\text{The sum of the normalized usage amount of Usage line items}}{\text{The normalized usage amount of the RIFee for size flexible Reserved Instances}}$. 전체 선결제 RI의 기본 요금 금액이 0보다 크지 않으므로 전체 선결제 RI의 값은 0입니다.	double
reservation_reservation_a_r_n	이 항목이 혜택을 받는 RI의 Amazon 리소스 이름(ARN)입니다. 이는 또한 'RI 리즈 ID'라고 합니다. 이는 이 특정 AWS 예약 인스턴스의 고유 식별자입니다. 값 문자열에는 AWS 서비스 이름과 RI를 구매한 리전도 포함됩니다.	string

열 이름	설명	데이터 유형
reservation_start_time	연결된 예약형 인스턴스 기간의 시작 날짜입니다.	string
reservation_subscription_id	품목과 관련 제안을 매핑하는 고유한 식별자입니다. RI ARN을 AWS 예약 인스턴스의 식별자로 사용할 것을 권장하지만 둘 다 사용할 수 있습니다.	string
reservation_total_reserved_normalized_units	예약 구독의 모든 인스턴스에 대한 총 예약 정규화 단위 수입니다. 예를 곱하여 총 정규화 단위 reservation/NormalizedUnitsPerReservation 를 AWS 계산합니다 reservation/NumberOfReservations .	string
reservation_total_reserved_units	TotalReservedUnits 은 Fee 및 RIFee 품목을 둘 다 채웁니다.	string
reservation_units_per_reservation	UnitsPerReservation 은 Fee 및 RIFee 품목을 둘 다 채웁니다.	string

열 이름	설명	데이터 유형
reservation_unused_amortized_upfront_fee_for_billing_period	청구 기간에 대한 선결제 상각 열은 전체 선결제 RI 및 부분 선결제 RI에 대한 초기 선결제 요금의 일부를 상각합니다. 선결제 없음 RI의 경우 선결제 금액이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.	double
reservation_unused_normalized_unit_quantity	유연한 크기의 리전 RI에 대해 청구 기간 동안 사용하지 않은 미사용 정규화 유닛 수입니다.	double
reservation_unused_quantity	이 결제 기간 동안 사용하지 않은 RI 시간의 수.	double
reservation_unused_recurring_fee	부분 선결제 및 선결제 없음 RI에 대해 미사용 예약 시간과 연결된 기본 요금입니다. 전체 선결제 RI의 기본 요금이 0보다 크지 않으므로 전체 선결제 RI의 값은 0입니다.	double
reservation_upfront_value	AWS 예약 인스턴스에 대해 지불된 선결제 가격입니다. 선결제 없음 RI의 경우, 이 값은 0입니다.	double

리소스 태그 열

리소스 태그 열에는 품목에 적용되는 리소스 태그에 대한 데이터가 포함됩니다. 리소스 태그도 태그 열에 포함되므로 태그 열을 선택한 경우 이 열을 선택할 필요가 없습니다.

열 이름	설명	데이터 유형
resource_tags	<p>주어진 품목에 대한 리소스 태그와 해당 값의 키-값 쌍이 포함된 맵 열입니다. 이 열의 값은 모두 데이터 유형이 '문자열'입니다.</p> <p>리소스 태그 키는 Billing 콘솔에서 비용 할당 태그로 활성화된 경우에만 이 열에 표시됩니다. 활성화한 후에는 특정 품목에 적용되는 값이 있는 경우에만 키가 맵 열에 표시됩니다.</p> <p>점 연산자를 사용하여 이 열의 키를 개별 열로 쿼리할 수 있습니다. 자세한 내용은 데이터 쿼리를 참조하세요.</p>	map <string, string>

절감형 플랜 열

절감형 플랜 열에는 품목에 적용되는 절감형 플랜에 대한 데이터가 포함됩니다.

열 이름	설명	데이터 유형
savings_plan_amortized_upfront_commitment_for_billing_period	청구 기간 동안 부과되는 절감형 플랜 구독에 대한 선결제 금액입니다. 이번 달 동안 분할 상환되는 전체 선결제 절감형 플랜 및 부분 선결제 절감형 플	double

열 이름	설명	데이터 유형
	랜의 초기 선결제 금액입니다. 선결제 없음 절감형 플랜의 경우 값은 0입니다.	
savings_plan_end_time	절감형 플랜 계약의 만료 날짜입니다.	string
savings_plan_instance_type_family	특정 사용량과 연결된 인스턴스 패밀리입니다.	string
savings_plan_net_amortized_upfront_commitment_for_billing_period	청구 기간 동안의 절감형 플랜 구독 선결제 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
savings_plan_net_recurring_commitment_for_billing_period	절감형 플랜 요금의 혼합되지 않은 순 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
savings_plan_net_savings_plan_effective_cost	절감형 플랜의 실질 비용은 사용량을 요금으로 나눈 값입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	double
savings_plan_offering_type	구매한 절감형 플랜의 유형을 설명합니다.	string

열 이름	설명	데이터 유형
savings_plan_payment_option	절감형 플랜에 사용 가능한 결제 옵션입니다.	string
savings_plan_purchase_term	절감형 플랜의 기간 또는 약정을 설명합니다.	string
savings_plan_recurring_commitment_for_billing_period	절감형 플랜 구독에 대한 월간 기본 요금입니다. 예를 들어 부분 선결제 절감형 플랜 또는 선결제 없음 절감형 플랜에 대한 월간 기본 요금입니다.	double
savings_plan_region	서비스를 호스팅하는 AWS 리전(지리적 영역)입니다 AWS . 이 필드를 사용하여 특정 AWS 리전의 지출을 분석할 수 있습니다.	string
savings_plan_savings_plan_ar_n	고유한 절감형 플랜 식별자입니다.	string
savings_plan_savings_plan_effective_cost	절감형 플랜 월간 약정 금액(선결제 및 기본 요금)의 비율은 각 사용량 행에 할당됩니다.	double
savings_plan_savings_plan_rate	사용량에 대한 절감형 플랜 요금입니다.	double
savings_plan_start_time	절감형 플랜 계약의 시작 날짜입니다.	string

열 이름	설명	데이터 유형
savings_plan_total_commitment_to_date	해당 시간에 대한 지금까지의 총 상각된 선결제 약정 및 반복 청구 요금 약정입니다.	double
savings_plan_used_commitment	사용된 절감형 플랜 약정의 총 금액(달러)입니다. (SavingsPlanRated와 사용량을 곱함)	double

분할 품목 열

split_line_item 헤더 아래에 있는 열은 비용 할당 데이터 분할 기능을 선택한 경우 데이터 내보내기에 나타나는 필드입니다. 자세한 내용은 [분할 비용 할당 데이터 이해](#)를 참조하세요. 이 기능은 Amazon ECS(Fargate 포함) AWS Batch 및 Amazon EKS로만 제한됩니다.

열 이름	설명	데이터 유형
split_line_item_actual_usage	테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨 Amazon ECS 작업 또는 Kubernetes 포드를 위해 지정된 기간 동안 발생한 vCPU, 메모리, 액셀러레이터 리소스 사용량(lineItem/UsageType 기준)입니다.	double
split_line_item_net_split_cost	테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨 모든 할인이 적용된 후의 Amazon ECS 작업 또는	double

열 이름	설명	데이터 유형
	Kubernetes 포드에 대한 실질 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.	
split_line_item_net_unused_cost	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>모든 할인이 적용된 후의 Amazon ECS 작업 또는 Kubernetes 포드에 대한 실질 미사용 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.</p>	double

열 이름	설명	데이터 유형
split_line_item_parent_resource_id	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Amazon EKS 포드와 연결된 상위 EC2 인스턴스의 리소스 ID입니다(linItem/ResourceId 열에서 참조). 상위 리소스 ID는 지정된 기간 동안의 ECS 작업 또는 Kubernetes 포드 워크로드가 상위 EC2 인스턴스에서 실행되었음을 의미합니다. 이는 EC2 시작 유형의 Amazon ECS 작업 또는 Kubernetes 포드에만 적용됩니다.</p>	string
split_line_item_public_on_demand_split_cost	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>퍼블릭 온디맨드 인스턴스 요금(pricing/publicOnDemandRate 열에서 참조)을 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리 비용(linItem/UsageType 기준)입니다.</p>	double

열 이름	설명	데이터 유형
split_line_item_public_on_demand_unused_cost	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>퍼블릭 온디맨드 인스턴스 요금을 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리의 미사용 비용 (lineitem/UsageType 기준)입니다. 미사용 비용은 지정된 기간 동안 사용되지 않은 EC2 인스턴스(splitLineitem/ParentResourceIdcolumn 열에서 참조)의 리소스(CPU 또는 메모리)와 관련된 비용입니다.</p>	double
split_line_item_reserved_usage	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Kubernetes 포드에 대해 지정된 기간 동안 구성한 vCPU, 메모리, 액셀러레이터 리소스 사용량(lineitem/UsageType 기준)입니다.</p>	double

열 이름	설명	데이터 유형
split_line_item_split_cost	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리 비용(linItem/UsageType 기준)입니다. 여기에는 EC2 인스턴스(splitLinItem/parentResourceId 열에서 참조)에 예약 또는 절감형 플랜에 대한 선결제 요금 또는 부분 선결제 요금이 있는 경우 분할 상환 비용이 포함됩니다.</p>	double
split_line_item_split_usage	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리 사용량(linItem/UsageType 기준)입니다. 이는 splitLinItem/ReservedUsage 또는 splitLinItem/ActualUsage의 최대 사용량으로 정의됩니다.</p>	double

열 이름	설명	데이터 유형
split_line_item_split_usage_ratio	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Kubernetes 포드에 할당된 vCPU, 메모리, 액셀러레이터 리소스(linItem/UsageType 기준)와 EC2 인스턴스에서 사용 가능한 전체 vCPU, 메모리, 액셀러레이터 리소스의 비율(splitLinItem/ParentResourceId 열에서 참조)입니다.</p>	double

열 이름	설명	데이터 유형
split_line_item_unused_cost	<p>테이블 구성: INCLUDE SPLIT COST ALLOCATION DATA에 의해 추가됨</p> <p>Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU, 메모리, 액셀러레이터 리소스의 미사용 비용(lineItem/UsageType 기준)입니다. 미사용 비용은 지정된 기간 동안 사용되지 않은 EC2 인스턴스(splitLineItem/ParentResourceId 열에서 참조)의 리소스(vCPU, 메모리, 액셀러레이터 리소스)와 관련된 비용입니다. 여기에는 EC2 인스턴스(splitLineItem/parentResourceId)에 예약 또는 절감형 플랜에 대한 선결제 요금 또는 부분 선결제 요금이 있는 경우 분할 상환 비용이 포함됩니다.</p>	double

태그 열

태그 열에는 항목에 적용되는 사용자, 계정, 비용 범주 및 리소스 태그에 대한 데이터가 포함됩니다. 이 열을 선택하는 경우 CUR 2.0에서 리소스 태그 및 비용 범주 열을 선택할 필요가 없습니다.

열 이름	설명	데이터 유형
tags	모든 태그의 키-값 페어와 지정된 항목에 대한 해당 값을 포함하는 맵 열입니다. 이 열의 값은	map <string, string>

열 이름	설명	데이터 유형
	<p>모두 데이터 유형이 '문자열'입니다.</p> <p>태그 키는 결제 콘솔에서 비용 할당 태그로 활성화된 경우에만 열에 표시됩니다. 활성화 후에는 특정 품목에 적용되는 값이 있는 경우에만 키가 맵 열에 표시됩니다.</p> <p>점 연산자를 사용하여 이 열의 키를 개별 열로 쿼리할 수 있습니다. 자세한 내용은 데이터 쿼리를 참조하세요.</p>	

태그 접두사 이해 및 태그 키 중첩

다른 AWS 태그 지정 메커니즘과 함께 비용 할당 태그를 사용하는 경우 다른 태그 지정 컨텍스트에서 동일한 태그 키(예: "department" 또는 "aws:createdBy")가 나타나는 상황이 발생할 수 있습니다. 충돌을 방지하고 정확한 비용 할당을 보장하기 위해 이러한 태그에 AWS 자동으로 접두사를 붙입니다.

태그 접두사 유형

AWS 는 다음 접두사를 사용하여 다양한 태그 소스를 구분합니다.

1. resourceTags/#- AWS 리소스에 직접 적용되는 태그입니다.
2. userAttribute/#- IAM Identity Center에서 가져온 사용자 속성입니다.
3. accountTag/#- AWS 계정 수준에서 적용된 태그입니다.
4. costCategory/#- AWS Cost Categories에서 파생된 태그입니다.
5. iamPrincipal/- [IAM 보안 주체에 적용된 태그입니다.](#)

예: 태그 키 중첩

여러 태그 지정 메커니즘이 동일한 태그 키를 사용하는 시나리오를 생각해 보세요. 가 이를 처리하는 방법은 AWS 다음과 같습니다.

```
{
  "resourceTags/department": "teamA",
  "resourceTags/appName": "app1",
  "userAttribute/Department": "teamB",
  "accountTag/department": "teamC",
  "accountTag/appName": "app3",
  "costCategory/department": "teamD"
}
```

이 예시는 다음과 같이 설정되어 있습니다.

- 리소스는 리소스 수준에서 부서 "teamA"로 태그가 지정됩니다.
- 리소스에 액세스한 사용자는 IAM Identity Center의 "teamB" 부서에 속합니다.
- AWS 계정에 "teamC" 부서를 나타내는 계정 수준 태그가 있습니다.
- 비용 범주 규칙이 비용을 "teamD" 부서에 할당했습니다.

각 태그는 고유한 접두사로 보존되므로 여러 관점에서 동시에 비용을 분석할 수 있습니다. 이를 통해 다음과 같은 작업이 가능합니다.

- 어떤 리소스가 어떤 팀에 속하는지 추적(resourceTags/department)
- 리소스를 소비하는 부서의 사용자 이해(userAttribute/Department)
- 계정 소유권에 따라 비용 할당(accountTag/department)
- 비용 범주를 통해 사용자 지정 비즈니스 로직 적용(costCategory/department)

용량 예약 열

용량 예약 열에는 항목에 적용되는 용량 예약에 대한 데이터가 포함되어 있습니다.

열 이름	설명	데이터 형식	Nullability	속성
capacity_reservation_arn	테이블 구성: 추가자: 용량 예약 데이터 포함	String	Nullable	요금이 용량 예약과 관련된 경우 이 필드는 null이 아닙니다.

열 이름	설명	데이터 형식	Nullability	속성
	용량 예약 ARN은 용량 예약의 고유 식별자를 나타냅니다.			<p>요금이 용량 예약의 미사용 부분을 나타내는 경우이 필드는 null이 아닙니다.</p> <p>요금이 용량 예약과 관련이 없는 경우이 필드는 null입니다.</p>

열 이름	설명	데이터 형식	Nullability	속성
capacity_reservation_status	<p>테이블 구성: 추가자: 용량 예약 데이터 포함</p> <p>행 항목이 capacity_reservation_arn 열에 식별된 용량 예약의 소비를 나타내는지 여부 또는 용량 예약이 사용되지 않은 경우 또는 용량 예약이 예약된 경우를 나타냅니다.</p>	String	Nullable	<p>capacity_reservation_arn이 null인 경우 이 필드는 null입니다.</p> <p>capacity_reservation_arn이 null이 아니고 line_item_line_item_type이 Usage 또는 SavingsPlannedCoveredUsage 또는 DiscountedUsage인 경우 이 필드는 null이 아닙니다.</p> <p>이 필드에는 허용된 값인 예약됨, 사용됨 또는 미사용 중 하나가 포함됩니다.</p>

열 이름	설명	데이터 형식	Nullability	속성
capacity_reservation_type	<p>테이블 구성: 추가자: 용량 예약 데이터 포함</p> <p>용량 예약 유형 필드는 구매한 용량 예약 유형을 나타냅니다. 현재 ML에는 2가지 유형의 ODCR 및 EC2 용량 블록이 있습니다.</p>	String	Nullable	<p>capacity_reservation_arn이 null인 경우 이 필드는 null입니다.</p> <p>capacity_reservation_arn이 null이 아니고 line_item_line_item_type이 Usage 또는 SavingsPlanCoveredUsage 또는 DiscountedUsage인 경우 이 필드는 null이 아닙니다.</p> <p>이 필드에는 ML에 허용되는 값인 ODCR 또는 EC2 용량 블록 중 하나가 포함되어 있습니다.</p>

비용 최적화 권장 사항(Cost Optimization Hub에서)

비용 최적화 권장 사항 테이블에는 Cost Optimization Hub의 비용 최적화 권장 사항이 포함되어 있습니다. Cost Optimization Hub 권장 사항은 AWS Compute Optimizer에서 통합되며 리소스 크기 조정, 유휴 리소스 삭제, Savings Plans, 예약 인스턴스 등 15가지 이상의 최적화 유형으로 구성됩니다. 자세한 정보는 AWS 비용 관리 사용 설명서의 [Cost Optimization Hub](#)를 참조하세요.

비용 최적화 권장 사항의 SQL 테이블 이름은 COST_OPTIMIZATION_RECOMMENDATIONS입니다.

테이블 구성

테이블 구성은 데이터 내보내기에서 테이블을 쿼리하기 전에 사용자가 테이블의 데이터나 스키마를 변경하도록 설정할 수 있는 사용자 제어 속성입니다. 테이블 구성은 JSON 문으로 저장되며 AWS SDK/CLI의 사용자 입력 또는 콘솔의 사용자 선택을 통해 지정됩니다.

비용 최적화 권장 사항에는 다음과 같은 테이블 구성이 있습니다.

구성 이름	설명	유효값
INCLUDE_ALL_RECOMMENDATIONS	<p>‘FALSE’로 설정하면 서로 호환되지 않는 모든 권장 사항 세트에서 절감 가치가 가장 높은 권장 사항만 테이블에 유지됩니다. 예를 들어 인스턴스를 종료하라는 권장 사항과 동일한 인스턴스의 크기를 조정하라는 권장 사항 중에서 ‘인스턴스 종료’만 유지됩니다.</p> <p>‘TRUE’로 설정하면 모든 권장 사항이 테이블에 유지됩니다.</p> <p>이는 Cost Optimization Hub 콘솔에서 그룹 관련 권장 사항이라고도 합니다. 자세한 내용은 AWS 비용 관리 사용 설명서의 관련 권장 사항 그룹화를 참조하세요.</p>	TRUE, FALSE

구성 이름	설명	유효값
FILTER	<p>이를 통해 다양한 권장 사항 특성을 기반으로 권장 사항을 필터링할 수 있습니다. 절감 중복 제거 알고리즘이 적용되기 전에 필터가 테이블에 적용됩니다.</p> <p>Cost Optimization Hub 콘솔과 동일한 파라미터를 사용하여 필터링할 수 있습니다. 자세한 내용은 AWS 비용 관리 사용 설명서의 비용 최적화 기회 우선 순위 지정을 참조하세요.</p> <p>필터 설명은 Cost Optimization Hub list-recommendations API의 filter 파라미터에 사용된 것과 동일한 JSON 구조를 사용하여 이 구성에 제공됩니다. JSON 문자열로 제공되어야 합니다. 자세한 내용은 list-recommendations 구조를 참조하세요.</p>	<p>Cost Optimization Hub list-recommendations API에 유효한 JSON 문자열입니다.</p>

서비스 연결 역할

비용 최적화 권장 사항 테이블의 내보내기를 생성하려면 데이터 내보내기의 서비스 연결 역할이 필요합니다. 서비스 연결 역할을 생성하는 방법에 대한 자세한 내용은 AWS 비용 관리 사용 설명서의 [데이터 내보내기 서비스 연결 역할](#)을 참조하세요.

AWS 조직 지원

Cost Optimization Hub는 AWS Organizations와 통합되어 관리 계정이 Cost Optimization Hub에서 멤버 계정 권장 사항을 볼 수 있는지 여부를 제어합니다. 자세한 내용은 AWS 비용 관리 사용 설명서의 [Cost Optimization Hub 시작하기](#)를 참조하세요.

지정된 AWS 계정에 대한 비용 최적화 권장 사항 표는 Cost Optimization Hub에 대해 구성한 것과 동일한 AWS Organizations 설정을 상속합니다. 즉, 비용 최적화 권장 사항 테이블의 내용은 각 AWS 계정의 Cost Optimization Hub에 표시되는 권장 사항과 일치합니다.

비용 최적화 권장 사항 열

열 이름	설명	데이터 유형	Null 값 허용됨
account_id	권장 사항이 적용되는 계정 ID입니다.	string	아니요
account_name	권장 사항이 적용되는 계정 이름입니다.	string	아니요
action_type	권장 사항을 채택하여 수행할 수 있는 작업 유형입니다.	string	아니요
currency_code	권장 사항에 사용되는 통화 코드입니다.	string	아니요
current_resource_details	JSON 문자열 형식의 리소스에 대한 세부 정보입니다.	string	예
current_resource_summary	현재 리소스에 대한 설명입니다.	string	예
current_resource_type	리소스의 유형입니다.	string	예
estimated_monthly_cost_after_discount	할인 후 현재 리소스의 예상 월별 비용입니다. 예약형 인스턴스 및 절	double	예

열 이름	설명	데이터 유형	Null 값 허용됨
	감형 플랜의 경우 적격 사용 비용을 나타냅니다.		
estimated_monthly_cost_before_discount	할인 전 현재 리소스의 예상 월별 비용입니다. 예약형 인스턴스 및 절감형 플랜의 경우 적격 사용 비용을 나타냅니다.	double	아니요
estimated_monthly_savings_after_discount	할인 후 권장 사항의 예상 월별 절감액입니다.	double	예
estimated_monthly_savings_before_discount	할인 전 권장 사항의 예상 월별 절감액입니다.	double	아니요
estimated_savings_percentage_after_discount	비용 계산 록백 기간 동안의 총 비용 대비 할인 후 예상 절감률입니다.	double	예
estimated_savings_percentage_before_discount	비용 계산 록백 기간 동안의 총 비용 대비 할인 전 예상 절감률입니다.	double	아니요
implementation_effort	권장 사항을 구현하는데 필요한 노력입니다.	string	아니요
last_refresh_timestamp	권장 사항이 마지막으로 생성된 시간입니다.	timestamp	아니요
recommendation_ID	권장 사항의 ID입니다.	string	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
recommendation_loo kback_period_in_days	권장 사항을 생성하는 데 사용되는 록백 기간입니다.	integer	아니요
recommendation_sou rce	권장 사항의 소스입니다.	string	아니요
recommended_resour ce_details	JSON 문자열 형식의 권장 리소스에 대한 세부 정보입니다.	string	예
recommended_resour ce_summary	권장 리소스에 대한 설명입니다.	string	예
recommended_resour ce_type	권장 사항의 리소스 유형입니다.	string	예
region	리소스의 AWS 리전입니다.	string	예
resource_arn	리소스의 Amazon 리소스 이름(ARN)입니다.	string	예
restart_needed	권장 사항을 구현하려면 다시 시작해야 하는지 여부를 나타냅니다.	boolean	아니요
rollback_possible	권장 사항의 구현을 롤백할 수 있는지 여부를 나타냅니다.	boolean	아니요
tags	권장 사항이 있는 리소스와 연결된 태그 목록입니다.	map	예

AWS 열이 있는 FOCUS 1.2

AWS 열이 있는 FOCUS 1.2 테이블에는 FinOps Open Cost and Usage Specification(FOCUS) 1.0 형식의 비용 및 사용량 데이터와 독점 결제 데이터가 포함된의 추가 열 AWS 3개가 포함되어 있습니다. 이러한 열은 `x_Discounts`, `x_Operation` 및 `x_ServiceCode`입니다. FOCUS 오픈 소스 사양의 자세한 내용은 [FOCUS](#) 웹사이트를 참조하세요.

AWS 열이 있는 FOCUS 1.2의 SQL 테이블 이름은 `FOCUS_1_2_AWS`입니다.

테이블 구성

테이블 구성은 데이터 내보내기에서 테이블을 쿼리하기 전에 사용자가 테이블의 데이터나 스키마를 변경하도록 설정할 수 있는 사용자 제어 속성입니다. 테이블 구성은 JSON 문으로 저장되며 AWS SDK/CLI의 사용자 입력 또는 콘솔의 사용자 선택을 통해 지정됩니다.

FOCUS 1.2에는 다음과 같은 테이블 구성이 있습니다.

구성 이름	설명	유효값
TIME_GRANULARITY	이 구성은 FOCUS 1.2 테이블의 비용 및 사용량 항목을 서로 다른 시간 단위로 변경합니다. 예를 들어, 'HOURLY'를 선택하면 모든 품목이 한 시간의 사용량을 나타냅니다.	시간당, 매일, 월별

AWS 조직 지원

AWS 열이 있는 FOCUS 1.2 테이블은 AWS Organizations의 통합 결제 기능에서 수행한 설정을 상속합니다. 통합 결제가 활성화되면 관리 계정과 멤버 계정에 대한 다양한 동작이 존재합니다. 관리 계정을 사용하는 경우 AWS 열이 있는 FOCUS 1.2 테이블에는 관리 계정 및 조직의 모든 멤버 계정에 대한 비용 및 사용 데이터가 포함됩니다. 멤버 계정을 사용하는 경우 AWS 열 테이블이 있는 FOCUS 1.2에는 해당 멤버 계정의 비용 및 사용 데이터만 포함됩니다.

조직에 가입한 후 멤버 계정은 해당 계정이 조직의 멤버로 있었던 기간 동안의 데이터만 내보낼 수 있습니다. 매월 15일에 조직 A를 떠나 조직 B에 가입하는 멤버 계정을 예로 들어 보겠습니다. 그런 다음

멤버 계정이 보고서를 생성합니다. 멤버 계정은 조직 B에 가입한 후 내보내기를 생성했으므로 멤버 계정의 해당 월의 AWS 열이 있는 FOCUS 1.2 내보내기에는 해당 계정이 조직 B의 멤버였던 기간의 비용 및 사용 데이터만 포함됩니다.

AWS 열이 있는 FOCUS 1.2

열 이름	설명	데이터 유형	Null 값 허용됨
AvailabilityZone	고가용성 및 내결함성을 제공하는 리전 내에서 물리적으로 분리되고 격리된 영역에 공급자가 할당한 식별자입니다.	string	예
BilledCost	선결제 요금 분할 상환(일회성 또는 반복성)을 제외한 모든 할인 요금 및 할인을 포함하여 인보이스 발행의 기준으로 사용되는 요금입니다.	double	아니요
BillingAccountId	공급자가 결제 계정에 할당한 식별자입니다.	string	아니요
BillingAccountName	결제 계정에 할당된 표시 이름입니다.	string	예
BillingAccountType	결제 계정 유형을 식별하기 위해 공급자가 할당한 이름입니다.	string	아니요
BillingCurrency	요금이 청구된 통화를 나타냅니다.	string	아니요
BillingPeriodEnd	결제 기간의 배타적 종료 제한입니다.	timestamp_milliseconds	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
BillingPeriodStart	결제 기간의 포함 시작 경계입니다.	timestamp_milliseconds	아니요
CapacityReservationId	공급자가 용량 예약에 할당한 식별자입니다.	string	예
CapacityReservationStatus	요금이 용량 예약의 소비를 나타내는지 또는 용량 예약이 사용되지 않은 시기를 나타내는지 여부를 나타냅니다.	string	예
ChargeCategory	청구 방법의 특성에 따라 요금의 최상위 분류를 나타냅니다.	string	아니요
ChargeClass	행이 이전에 청구된 청구 기간에 대한 수정을 나타내는지 여부를 나타냅니다.	string	예
ChargeDescription	요금의 목적 및 가격에 대한 독립형 요약입니다.	string	예
ChargeFrequency	요금이 발생하는 빈도를 나타냅니다.	string	아니요
ChargePeriodEnd	요금 기간의 배타적 종료 제한입니다.	timestamp_milliseconds	아니요
ChargePeriodStart	요금 기간의 포함 시작 경계입니다.	timestamp_milliseconds	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
CommitmentDiscount Category	CommitmentDiscount Id 열에서 식별된 약정 할인이 사용량 또는 비용(일명 "지출")을 기반으로 하는지 여부를 나타냅니다.	string	예
CommitmentDiscount Id	공급자가 약정 할인에 할당한 식별자입니다.	string	예
CommitmentDiscount Name	약정 할인에 할당된 표시 이름입니다.	string	예
CommitmentDiscount Quantity	약정 할인 단위로 표시된 약정 할인 관련 행에서 구매하거나 계산한 약정 할인 금액입니다.	double	예
CommitmentDiscount Status	요금이 약정 할인 사용 또는 약정 금액의 미사용 부분에 해당하는지 여부를 나타냅니다.	string	예
CommitmentDiscount Type	행에 적용되는 약정 할인 유형에 대해 공급자가 할당한 식별자입니다.	string	예
CommitmentDiscount Unit	공급자가 약정 할인의 약정 할인 수량을 측정하는 방법을 나타내는 공급자 지정 측정 단위입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
ConsumedQuantity	사용된 단위를 기준으로 사용된 리소스 또는 서비스와 연결된 측정된 SKU의 볼륨입니다.	double	예
ConsumedUnit	공급자가 리소스 또는 서비스와 연결된 측정된 SKU의 사용량을 측정하는 방법을 나타내는 공급자 지정 측정 단위입니다.	string	예
ContractedCost	계약 단가와 해당 요금 수량을 곱하여 계산된 비용입니다.	double	아니요
ContractedUnitPrice	협상된 약정 할인 또는 기타 할인을 제외한 협상된 할인이 있는 경우 이를 포함하여 관련 SKU의 단일 요금 단위에 대해 합의된 단가입니다.	double	예
EffectiveCost	모든 할인 요금, 할인 및이 요금을 부담한 관련 선불 구매(일회성 또는 반복성)의 해당 부분을 적용한 후 부과되는 분할 상환 비용입니다.	double	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
InvoiceId	지정된 결제 계정의 해당 결제 기간에 일부 또는 모든 요금을 캡슐화하는 인보이스의 공급자 할당 식별자입니다.	string	예
InvoiceIssuerName	사용된 리소스 또는 서비스에 대한 인보이스 발행을 담당하는 엔터티의 이름입니다.	string	아니요
ListCost	정가와 해당 요금 수량을 곱하여 계산된 비용입니다.	double	아니요
ListUnitPrice	할인을 제외한 관련 SKU의 단일 요금 단위에 대해 제안된 공급자 게시 단가입니다.	double	예
PricingCategory	사용 또는 구매 시 요금에 사용되는 요금 모델을 설명합니다.	string	예
PricingCurrency	리소스 또는 서비스의 가격이 책정된 국가 또는 가상 통화 액면가입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
PricingCurrencyContractedUnitPrice	협상된 약정 할인 또는 기타 할인을 제외하고 존재하는 경우 협상된 할인을 포함하여 관련 SKU의 단일 요금 단위에 대해 합의된 단가이며 요금 통화로 표시됩니다.	double	예
PricingCurrencyEffectiveCost	요금 통화로 표시된 대로 모든 할인 요금, 할인 및이 요금을 적용한 관련 선불 구매(일회성 또는 반복성)의 해당 부분을 적용한 후 발생하는 요금 비용입니다.	double	예
PricingCurrencyListUnitPrice	할인을 제외하고 요금 통화로 표시된 관련 SKU의 단일 요금 단위에 대해 제안된 공급자 게시 단가입니다.	double	예
PricingQuantity	요금 단위에 따라 사용 또는 구매한 리소스 또는 서비스와 연결된 지정된 SKU의 볼륨입니다.	double	예

열 이름	설명	데이터 유형	Null 값 허용됨
PricingUnit	공급자가 블록 요금과 같은 요금 규칙을 적용한 후 사용량 및 구매한 수량을 측정한 요금을 나타내는 단위 가격을 결정하기 위한 공급자 지정 측정 단위입니다.	string	예
ProviderName	리소스 또는 서비스를 구매할 수 있게 만든 엔터티의 이름입니다.	string	아니요
PublisherName	구매한 리소스 또는 서비스를 생성한 엔터티의 이름입니다.	string	아니요
RegionId	리소스가 프로비저닝되거나 서비스가 제공되는 격리된 지리적 영역에 대한 공급자 할당 식별자입니다.	string	예
RegionName	리소스가 프로비저닝되거나 서비스가 제공되는 격리된 지리적 영역의 이름입니다.	string	예
ResourceId	공급자가 리소스에 할당한 식별자입니다.	string	예
ResourceName	리소스에 할당된 표시 이름입니다.	string	예
ResourceType	요금이 적용되는 리소스의 종류입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
ServiceCategory	서비스의 핵심 함수를 기반으로 한 서비스의 최상위 분류입니다.	string	아니요
ServiceName	공급자(예: 클라우드 가상 머신, SaaS 데이터베이스, 시스템 통합자의 전문 서비스)로부터 구매할 수 있는 상품입니다.	string	아니요
ServiceSubcategory	코어 함수를 기반으로 서비스에 대한 서비스 범주의 보조 분류입니다.	string	아니요
Skuld	특정 SKU(예: 정량화 가능한 상품 또는 서비스 상품)를 나타내는 공급자 지정 고유 식별자입니다.	string	예
SkuMeter	특정 SKU가 요금을 청구하여 측정하거나 측정하는 기능에 대해 설명합니다.	string	예
SkuPriceld	사용하거나 구매한 리소스 또는 서비스와 관련된 특정 SKU 가격을 나타내는 공급자 지정 고유 식별자입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
SkuPriceDetails	해당 SKU 가격 ID의 모든 인스턴스에 의미 있고 공통적인 SKU 가격 ID의 속성 세트입니다.	map <string, string>	예
SubAccountId	리소스 또는 서비스 그룹에 할당된 ID로, 액세스 및/또는 비용 관리에 주로 사용됩니다.	string	예
SubAccountName	리소스 또는 서비스 그룹에 할당된 이름으로, 액세스 및/또는 비용 관리에 주로 사용됩니다.	string	예
SubAccountType	하위 계정의 유형을 식별하기 위해 공급자가 할당한 이름입니다.	string	예
Tags	잠재적 공급자 정의 또는 사용자 정의 태그 평가를 설명하는 태그 소스에 할당된 태그 세트입니다.	map <string, string>	예
x_Discounts	이 항목에 적용되는 특정 할인의 키-값 쌍을 포함하는 맵 열입니다.	map <string, double>	예

열 이름	설명	데이터 유형	Null 값 허용됨
x_Operation	이 행 항목에서 다루는 특정 AWS 작업입니다. 이 항목의 구체적인 사용량에 대해 설명합니다. 예를 들어 RunInstances 값은 Amazon EC2 인스턴스 작업을 나타냅니다.	string	예
x_ServiceCode	측정된 제품의 코드입니다. 예를 들어 Amazon EC2는 Amazon Elastic Compute Cloud의 제품 코드입니다.	string	예

AWS 열 적합성 격차가 있는 FOCUS 1.2

다음 표에는 AWS 열이 있는 FOCUS 1.2 테이블의 내보내기에 존재할 수 있는 모든 적합성 격차가 나와 있습니다. 관련 시나리오에 대한 비용 및 사용 데이터를 받지 못할 경우 특정 적합성 격차가 내보내기에 적용되지 않습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.2 요구 사항	적합성 격차 설명
누락된 데이터	ContractedUnitPrice	ContractedUnitPrice는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	ContractedUnitPrice는 특정 제품 제안에서 null일 수 있습니다.
	InvoicelssuerName	InvoicelssuerName은 null이 아니어야 합니다.	InvoicelssuerName은 특정 요금에서 null일 수 있습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.2 요구 사항	적합성 격차 설명
	ListUnitPrice	ListUnitPrice는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	ListUnitPrice는 특정 제품 제안에서 null일 수 있습니다.
	PricingUnit	PricingUnit은 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	PricingUnit은 특정 제품 제안에서 null일 수 있습니다.
	PublisherName	PublisherName은 null이 아니어야 합니다.	PublisherName은 특정 요금에서 null일 수 있습니다.
	Skuld	Skuld는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	Skuld는 특정 제품 제안에서 null일 수 있습니다.
	SkuPriceld	SkuPriceld는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	SkuPriceld는 특정 항목에서 null이 아니어야 하지만 null일 수 있습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.2 요구 사항	적합성 격차 설명
	CapacityReservationStatus	CapacityReservationId가 null이 아니고 ChargeCategory가 Usage인 경우 CapacityReservationStatus는 null이 아니어야 합니다. CapacityReservationId ChargeCategory	CapacityReservationStatus는 예약 상태의 용량 예약에 대해 null일 수 있으며 "사용" 또는 "미사용"이 아닙니다.
잘못된 데이터	ConsumedQuantity	ConsumedQuantity는 실제 사용량을 표시하는 필수 열입니다.	ConsumedQuantity에는 요금이 청구된 사용량이 포함됩니다. 즉, 특정 서비스에 최소 요금 수량이 적용될 경우 ConsumedQuantity가 부정확할 수 있습니다. 예를 들어 Athena 쿼리의 최소값은 10MB이고 Glue 크롤러를 최소 10분 이상 실행해야 한다고 가정해 보겠습니다. 이러한 서비스의 경우 ConsumedQuantity는 최소 비용 부과 수량을 포함하는 값을 표시합니다.
	SkuMeter	Skuld가 null인 경우 SkuMeter는 null이어야 합니다. Skuld	SkuMeter는 특정 제품 제공에 대해 null일 수 있습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.2 요구 사항	적합성 격차 설명
	ConsumedUnit	ConsumedUnit nullability는 ConsumedQuantity와 일치해야 합니다.	ConsumedQuantity가 있는 특정 제품 제공의 경우 ConsumedUnit이 null일 수 있습니다. ConsumedQuantity

FOCUS 1.0에서 FOCUS 1.2로 마이그레이션

AWS Data Exports를 사용하면 AWS 열을 사용하여 FOCUS 1.2의 내보내기를 생성할 수 있습니다. 이 내보내기는 FOCUS 1.0과 동일한 표준화된 비용 및 사용 정보와 함께 인보이스 조정, 용량 예약 추적 및 SaaS 통합을 위한 몇 가지 개선 사항을 제공합니다. 그러나 FOCUS 1.2는 기존 행 수와 열 값에 영향을 미치는 주요 변경 사항을 도입합니다. 마이그레이션하기 전에 이러한 변경 사항을 주의 깊게 검토하세요.

FOCUS 1.2는 FOCUS 1.0에 비해 다음과 같은 개선 사항을 제공합니다.

- 인보이스 조정: FOCUS 1.2에는 간소화된 재무 마감 프로세스 및 인보이스 조정을 지원하는 InvoiceID 열이 포함되어 있습니다.
- 용량 예약 추적: FOCUS 1.2에는 미사용 용량 예약을 식별하고 추적하는 데 도움이 되는 CapacityReservationId 및 CapacityReservationStatus 열이 포함되어 있습니다.
- 가상 통화 지원: FOCUS 1.2에는 SaaS 공급자 비용 및 사용량 데이터를 FOCUS 1.2 형식으로 조인할 수 있는 새로운 요금 통화 열(PricingCurrency, PricingCurrencyEffectiveCost, PricingCurrencyListUnitPrice 및 PricingCurrencyContractedUnitPrice)이 포함되어 있습니다.

다음 표에는 FOCUS 1.2와 FOCUS 1.0의 차이점이 자세히 요약되어 있습니다.

기능	포커스 1.2	FOCUS 1.0
데이터 스키마	57개의 FOCUS 열 + 3 AWS 개의 열(총 60개)이 있는 FOCUS 1.2 사양입니다. 전체 열 목록은 FOCUS 1.2 with AWS columns을 참조하세요.	43개의 FOCUS 열 + 5개의 AWS 열(총 48개)이 있는 FOCUS 1.0 사양입니다. 전체 열 목록은 FOCUS 1.0 with AWS columns을 참조하세요.

기능	포커스 1.2	FOCUS 1.0
새로운 열	InvoiceIssuerId, CapacityReservationId, CapacityReservationStatus, CommitmentDiscountQuantity, CommitmentDiscountUnit, ServiceSubcategory, SkuMeter, SkuPriceDetails, PricingCurrency, PricingCurrencyEffectiveCost, PricingCurrencyListUnitPrice, PricingCurrencyContractedUnitPrice, BillingAccountType, SubAccountType	
열 제거	x_UsageType(이제 SkuMeter 열임) x_CostCategories(이제 'aws:tags:CostCategory/' 접두사가 붙은 태그 열에 포함됨)	
행 수	이제 ML 품목에 대한 온디맨드 용량 예약(ODCR) 및 EC2 용량 블록이 별도의 '사용됨' 및 '미사용됨' 상태로 분할됩니다. 이렇게 하면 행 수가 증가합니다.	용량 예약 상태 분석 세부 정보는 제공하지 않습니다.
PricingCategory 열 값	예약 인스턴스 및 Savings Plan 선결제 요금과 같은 약정 할인 구매는 이제 "표준"으로 분류됩니다.	"Committed"로 분류된 약정 할인 구매.
ConsumedQuantity 열 nullability	CommitmentDiscountStatus가 "미사용"일 때 Null입니다.	CommitmentDiscountStatus가 "Unused"인 경우 값은 "0"입니다.

기능	포커스 1.2	FOCUS 1.0
ConsumedUnit 열 nullability	ConsumedQuantity nullability에 연결됩니다(ConsumedQuantity가 null인 경우 null이어야 함).	독립적인 nullability 규칙.
태그 열 요구 사항	사용자 정의 및 공급자 정의 리소스 태그와 사용자 정의 비용 범주 태그를 포함합니다.	사용자 정의 및 공급자 정의 리소스 태그만 포함합니다.
파일 전송 대상	S3 버킷	S3 버킷
파일 출력 형식	GZIP, Parquet	GZIP, Parquet
SQL 테이블 이름	FOCUS_1_2_AWS	FOCUS_1_0_AWS
매니페스트 FocusVersion	"1.2"	"1.0"
테이블 구성	TIME_GRANULARITY를 시간당, 일별 또는 월별로 설정하도록 허용	는 시간당으로만 내보냅니다.

FOCUS 1.2의 스키마에 대한 자세한 내용은 섹션을 참조하세요 [AWS 열이 있는 FOCUS 1.2](#).

AWS 열이 있는 FOCUS 1.0

AWS 열이 있는 FOCUS 1.0 테이블에는 FinOps Open Cost and Usage Specification(FOCUS) 1.0 형식의 비용 및 사용량 데이터와 독점 결제 데이터가 포함된의 추가 열 AWS 5개가 포함되어 있습니다. 이러한 열은 x_CostCategories, x_Discounts, x_Operation, x_ServiceCode 및 x_UsageType입니다. FOCUS 오픈 소스 사양의 자세한 내용은 [FOCUS](#) 웹사이트를 참조하세요.

AWS 열이 있는 FOCUS 1.0의 SQL 테이블 이름은 입니다. FOCUS_1_0_AWS

테이블 구성

AWS 열이 있는 FOCUS 1.0 테이블에 대한 테이블 구성은 없습니다.

AWS 조직 지원

AWS 열이 있는 FOCUS 1.0 테이블은 AWS Organizations의 통합 결제 기능에서 수행한 설정을 상속합니다. 통합 결제가 활성화되면 관리 계정과 멤버 계정에 대한 다양한 동작이 존재합니다. 관리 계정을 사용하는 경우 AWS 열 포함 FOCUS 1.0 테이블에는 관리 계정 및 조직의 모든 멤버 계정에 대한 비용 및 사용 데이터가 포함됩니다. 멤버 계정을 사용하는 경우 AWS 열 테이블이 있는 FOCUS 1.0에는 해당 멤버 계정의 비용 및 사용 데이터만 포함됩니다.

조직에 가입한 후 멤버 계정은 해당 계정이 조직의 멤버로 있었던 기간 동안의 데이터만 내보낼 수 있습니다. 매월 15일에 조직 A를 떠나 조직 B에 가입하는 멤버 계정을 예로 들어 보겠습니다. 그런 다음 멤버 계정이 보고서를 생성합니다. 멤버 계정은 조직 B에 가입한 후 내보내기를 생성했으므로 멤버 계정의 월별 AWS 열이 있는 FOCUS 1.0 내보내기에는 해당 계정이 조직 B의 멤버였던 기간의 비용 및 사용 데이터만 포함됩니다.

AWS 열이 있는 FOCUS 1.0

열 이름	설명	데이터 유형	Null 값 허용됨
AvailabilityZone	고가용성 및 내결함성을 제공하는 리전 내에서 물리적으로 분리되고 격리된 영역에 공급자가 할당한 식별자입니다.	string	예
BilledCost	향후 적격 요금을 총당하기 위해 지불된 관련 구매의 상환을 제외한 모든 절감된 요금 및 할인을 포함하여 청구서 발행의 기초가 되는 요금입니다.	double	아니요
BillingAccountId	결제 계정의 공급자 할당 식별자입니다.	string	아니요
BillingAccountName	결제 계정의 공급자 할당 이름입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
BillingCurrency	리소스 또는 서비스에 대한 요금이 청구된 통화를 나타내는 식별자입니다.	string	아니요
BillingPeriodEnd	결제 기간이 종료되는 날짜 및 시간입니다.	timestamp_milliseconds	아니요
BillingPeriodStart	결제 기간이 시작되는 날짜 및 시간입니다.	timestamp_milliseconds	아니요
ChargeCategory	행이 선결제 또는 반복 요금, 이미 발생한 사용 비용, 사후 조정(예: 크레딧) 또는 세금을 나타내는지 여부를 나타내는 지표입니다.	string	아니요
ChargeClass	행이 정규 요금을 나타내는지 또는 하나 이상의 이전 요금을 수정한 것인지 여부를 나타내는 지표입니다.	string	예
ChargeDescription	추가 검색이 필요 없는 행의 상위 수준 컨텍스트입니다.	string	예
ChargeFrequency	요금이 발생하는 빈도를 나타내는 지표입니다.	string	아니요
ChargePeriodEnd	요금 부과 기간이 종료되는 날짜 및 시간입니다.	timestamp_milliseconds	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
ChargePeriodStart	요금 부과 기간이 시작 되는 날짜 및 시간입니다.	timestamp_milliseconds	아니요
CommitmentDiscount Category	CommitmentDiscount Id 열에 식별된 약정 기반 할인이 사용량 또는 비용(즉, 지출)을 기반으로 하는지 여부를 나타내는 지표입니다.	string	예
CommitmentDiscount Id	약정 기반 할인에 대해 공급자가 할당한 식별자입니다.	string	예
CommitmentDiscount Name	약정 기반 할인에 할당된 표시 이름입니다.	string	예
CommitmentDiscount Status	요금이 사용 또는 미사용 약정 할인에 해당하는지 여부를 나타내는 지표입니다.	string	예
CommitmentDiscount Type	행에 적용되는 약정 기반 할인 유형을 식별하기 위해 공급자가 할당한 이름입니다.	string	예
ConsumedQuantity	ConsumedUnit을 기반으로 사용 또는 구매한 특정 리소스 또는 서비스의 볼륨입니다.	double	예

열 이름	설명	데이터 유형	Null 값 허용됨
ConsumedUnit	공급자가 리소스 또는 서비스와 연결된 특정 SKU의 사용량을 측정하는 방법을 나타내는 공급자 할당 측정 단위입니다.	string	예
ContractedCost	ContractedUnitPrice와 해당 PricingQuantity를 곱하여 계산한 비용입니다.	double	아니요
ContractedUnitPrice	협상된 약정 기반 할인 또는 기타 할인을 제외한 모든 협상된 할인을 포함하여 관련 SKU의 단일 PricingUnit에 대해 합의된 단가입니다.	double	예
EffectiveCost	향후 적격 요금을 충당하기 위해 지불된 관련 구매(일회성 또는 반복성)의 상환으로 증가되는 모든 할인 요금 및 할인을 포함하는 비용입니다.	double	아니요
InvoiceIssuerName	사용된 소스 또는 서비스에 대해 인보이스를 발행할 책임이 있는 엔터티입니다. 일반적으로 비용 분석 및 보고 시나리오에 사용됩니다.	string	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
ListCost	ListUnitPrice와 해당 PricingQuantity를 곱하여 계산한 비용입니다.	double	아니요
ListUnitPrice	할인을 제외한 관련 SKU의 단일 PricingUnit에 대해 공급자가 게시한 추천 단가입니다.	double	예
PricingCategory	사용 또는 구매 시 요금에 사용되는 요금 모델입니다.	string	예
PricingQuantity	PricingUnit을 기반으로 사용 또는 구매한 리소스 또는 서비스와 연결된 특정 SKU의 볼륨입니다.	double	예
PricingUnit	공급자가 블록 요금과 같은 요금 규칙을 적용한 후 측정된 사용량 및 구매 수량의 비용을 계산하는 방법을 나타내는 단가 결정을 위한 공급자 할당 측정 단위입니다.	string	예
ProviderName	리소스 또는 서비스를 구매할 수 있도록 만든 엔터티입니다.	string	아니요
PublisherName	구매한 리소스 또는 서비스를 생성한 엔터티입니다.	string	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
RegionId	리소스가 프로비저닝되거나 서비스가 제공되는 격리된 지리적 영역에 대한 공급자 할당 식별자입니다.	string	예
RegionName	리소스가 프로비저닝되거나 서비스가 제공되는 격리된 지리적 영역의 이름입니다.	string	예
ResourceId	리소스에 대해 공급자가 할당한 식별자입니다.	string	예
ResourceName	리소스에 할당된 표시 이름입니다.	string	예
ResourceType	요금이 적용되는 리소스 유형입니다.	string	예
ServiceCategory	서비스의 핵심 함수를 기반으로 한 서비스의 최고 수준 분류입니다.	string	아니요
ServiceName	구매한 제안의 표시 이름입니다.	string	아니요
Skuld	하나 이상의 SKU 가격에서 일반적인 특성을 구성하는 공급자 지원 구성을 정의하는 고유 식별자입니다.	string	예
SkuPriceld	요금을 계산하는 데 사용되는 단가를 정의하는 고유 식별자입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
SubAccountId	리소스 또는 서비스 그룹에 할당된 ID로, 액세스 및/또는 비용 관리에 주로 사용됩니다.	string	예
SubAccountName	리소스 또는 서비스 그룹에 할당된 이름으로, 액세스 및/또는 비용 관리에 주로 사용됩니다.	string	예
Tags	잠재적 공급자 정의 또는 사용자 정의 태그 평가를 설명하는 태그 소스에 할당된 태그 세트입니다.	map <string, string>	예
x_CostCategories	주어진 품목에 대한 비용 범주와 해당 값의 키-값 쌍이 포함된 맵 열입니다.	map <string, string>	예
x_Discounts	이 항목에 적용되는 특정 할인의 키-값 쌍을 포함하는 맵 열입니다.	map <string, double>	예
x_Operation	이 행 항목에서 다루는 특정 AWS 작업입니다. 이 항목의 구체적인 사용량에 대해 설명합니다.	string	예
x_ServiceCode	이 항목에 사용되는 서비스의 코드입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
x_UsageType	이 항목의 사용량 세부 정보입니다.	string	예

AWS 열 적합성 격차가 있는 FOCUS 1.0

다음 표에는 AWS 열이 있는 FOCUS 1.0 테이블의 내보내기에 존재할 수 있는 모든 적합성 격차가 나와 있습니다. 관련 시나리오에 대한 비용 및 사용 데이터를 받지 못할 경우 특정 적합성 격차가 내보내기에 적용되지 않습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.0 요구 사항	적합성 격차 설명
누락된 데이터	ContractedUnitPrice	ContractedUnitPrice는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	ContractedUnitPrice는 특정 제품 제안에서 null일 수 있습니다.
	InvoiceIssuerName	InvoiceIssuerName은 null이 아니어야 합니다.	InvoiceIssuerName은 특정 요금에서 null일 수 있습니다.
	ListUnitPrice	ListUnitPrice는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	ListUnitPrice는 특정 제품 제안에서 null일 수 있습니다.
	PricingUnit	PricingUnit은 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	PricingUnit은 특정 제품 제안에서 null일 수 있습니다.

적합성 격차 유형	영향을 받는 열	FOCUS 1.0 요구 사항	적합성 격차 설명
	PublisherName	PublisherName은 null이 아니어야 합니다.	PublisherName은 특정 요금에서 null일 수 있습니다.
	Skuld	Skuld는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	Skuld는 특정 제품 제안에서 null일 수 있습니다.
	SkuPriceld	SkuPriceld는 수정 항목이 아닌 사용 및 구매 ChargeCategory 항목에서 null이 아니어야 합니다.	SkuPriceld는 특정 항목에서 null이 아니어야 하지만 null일 수 있습니다.
잘못된 데이터	ConsumedQuantity	ConsumedQuantity는 실제 사용량을 표시하는 필수 열입니다.	<p>ConsumedQuantity에는 요금이 청구된 사용량이 포함됩니다. 즉, 특정 서비스에 최소 요금 수량이 적용될 경우 ConsumedQuantity가 부정확할 수 있습니다.</p> <p>예를 들어 Athena 쿼리의 최소값은 10MB이고 Glue 크롤러를 최소 10분 이상 실행해야 한다고 가정해 보겠습니다. 이러한 서비스의 경우 ConsumedQuantity는 최소 비용 부과 수량을 포함하는 값을 표시합니다.</p>

Cost and Usage Dashboard

Cost and Usage Dashboard의 SQL 테이블 이름은 COST_AND_USAGE_DASHBOARD입니다.

테이블 구성

테이블 구성은 데이터 내보내기에서 테이블을 쿼리하기 전에 사용자가 테이블의 데이터나 스키마를 변경하도록 설정할 수 있는 사용자 제어 속성입니다.

Cost and Usage Dashboard에는 수정 가능한 테이블 구성이 없습니다.

AWS 조직 지원

비용 및 사용량 대시보드 테이블은 CUR 2.0 데이터에서 생성됩니다. 즉, CUR 2.0에 적용되는 것과 동일한 AWS Organizations 설정을 상속하고 동일한 동작이 적용됩니다. AWS Organizations가 비용 및 사용 대시보드에 적용되는 방식을 이해하려면 CUR 2.0 섹션에서 [AWS Organizations 지원](#)을 참조하세요.

Cost and Usage Dashboard 열

열 이름	설명
amortized_cost	<p>청구 기간 동안 선불 및 월간 예약 요금의 유효 비용을 반영합니다. 이 값은 품목 유형에 따른 비용 합계입니다. 비용은 다음과 같이 결정합니다.</p> <ul style="list-style-type: none"> • 품목 유형이 'SavingsPlanCoveredUsage'인 경우 비용은 절감형 플랜의 유효 비용입니다. • 품목 유형이 'SavingsPlanRecurringFee'인 경우 비용은 절감형 플랜 당일까지의 총 약정에서 사용한 약정을 뺀 값입니다. • 품목 유형이 'SavingsPlanNegation' 또는 'SavingsPlanUpfrontFee'인 경우, 비용은 0입니다. •

열 이름	설명
	<p>품목 유형이 'DiscountedUsage'인 경우 비용은 예약의 실질 비용입니다.</p> <ul style="list-style-type: none"> • 품목 유형이 'RIFee'인 경우 비용은 청구 기간 동안 상각된 미사용 선결제 요금과 예약의 미사용 기본 요금의 합계입니다. • 품목 유형이 'Fee'이고 예약 ARN이 있는 경우 비용은 0입니다. • 다른 모든 품목 유형의 경우 비용은 해당 품목의 비혼합 비용입니다.
availability_zone	이 항목을 호스팅하는 가용 영역입니다. us-east-1a 또는 us-east-1b를 예로 들 수 있습니다.
billing_entity	<p>인보이스 또는 트랜잭션이 AWS Marketplace용인지 아니면 다른 AWS 서비스 구매용인지 식별하는 데 도움이 됩니다. 가능한 값은 다음과 같습니다.</p> <p>AWS: Marketplace 이외의 AWS 서비스에 대한 트랜잭션을 AWS 식별합니다.</p> <p>AWS Marketplace: AWS Marketplace에서의 구매를 식별합니다.</p>
billing_period	<p>이 대시보드에 표시되는 청구 기간 시작일(UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ입니다.</p> <p>예: 2023-10-01T00:00:00.000Z</p>

열 이름	설명
charge_category	<p>이 품목에 적용되는 요금 범주입니다. 사용 가능한 범주는 다음과 같습니다.</p> <p>Running_Usage: charge_type이 'DiscountedUsage', 'SavingsPlanCoveredUsage', 'Usage' 중 하나인 경우</p> <p>Non_Usage: charge_type 아래에 있는 다른 모든 항목의 경우</p>
charge_type	<p>이 항목에 적용되는 요금 유형입니다. 가능한 모든 값은 여기에 있는 lineltem/LineltemType을 참조하세요.</p>
current_generation	<p>Amazon RDS 인스턴스가 현재 세대인지 아닌지를 식별하는 데 도움이 됩니다.</p>
database_engine	<p>데이터베이스의 데이터베이스 엔진을 설명합니다.</p> <p>예: PostgreSQL, Oracle</p>

열 이름	설명
instance_type	<p>인스턴스의 CPU, 네트워킹 및 스토리지 용량을 정의하는 인스턴스 유형, 크기 및 패밀리를 설명합니다.</p> <p>예: t2.small, m4.xlarge, t2.micro, m4.large, t2.large</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon EC2 • Amazon RDS • OpenSearch Service • Amazon ElastiCache • Amazon EMR • 전체 서비스 목록을 보려면 Column_Attribute_Service.zip을 다운로드하세요.
instance_type_family	<p>주어진 사용량과 연결된 인스턴스 패밀리입니다.</p> <p>예: t2, m4, m3</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon DocumentDB • Amazon RDS
invoice_id	<p>특정 항목과 연결된 ID입니다. 보고서가 완성될 때까지 invoice_id는 비어 있습니다.</p>

열 이름	설명
item_description	<p>항목 유형의 설명입니다. 예를 들어 사용량 항목의 설명은 지정된 기간 동안 어떤 유형의 사용량이 발생했는지 요약합니다.</p> <p>유연한 크기의 RI의 경우 설명은 그 혜택이 적용되는 RI와 일치합니다. 예를 들어 품목이 t2.micro에 해당하고 t2.small RI가 사용량에 적용된 경우 lineitem/LineitemDescription에 t2.small이 표시됩니다.</p> <p>RI 할인이 포함된 사용량 항목의 설명에는 항목에 포함되는 요금제를 포함합니다.</p>
legal_entity	<p>특정 제품 또는 서비스의 등록 판매자. 대부분의 경우 인보이스 발행 주체와 법인은 동일합니다. 타사 AWS Marketplace 트랜잭션의 경우 값이 다를 수 있습니다. 가능한 값은 다음과 같습니다.</p> <p>Amazon Web Services, Inc.: AWS 서비스를 판매하는 기업입니다.</p> <p>Amazon Web Services India Private Limited: 인도에서 AWS 서비스의 리셀러 역할을 하는 인도 현지 법인입니다.</p>
linked_account_id	<p>이 항목에서 사용된 계정의 계정 ID입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다. 이 필드를 사용하여 계정별 비용과 사용량을 추적할 수 있습니다.</p>
linked_account_name	<p>이 품목을 사용한 계정의 이름입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다. 이 필드를 사용하여 계정별 비용과 사용량을 추적할 수 있습니다.</p>

열 이름	설명
operation	이 행 항목에서 다루는 특정 AWS 작업입니다. 이 항목의 구체적인 사용량에 대해 설명합니다. 예를 들어 RunInstances 값은 Amazon EC2 인스턴스 작업을 나타냅니다.
payer_account_id	지급 계정의 계정 ID입니다. AWS Organizations에 속한 조직의 경우 이 항목은 관리 계정의 계정 ID입니다.
payer_account_name	지급 계정의 계정 이름입니다. AWS Organizations에 속한 조직의 경우 이 항목은 관리 계정의 이름입니다.
platform	<p>Amazon EC2 인스턴스의 운영 체제를 설명합니다.</p> <p>예: Amazon Linux, Ubuntu, Windows Server, Oracle Linux, FreeBSD</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon 앱Stream • Amazon EC2 • Amazon GameLift • Amazon Lightsail • Amazon WorkSpaces • Amazon CodeBuild
pricing_unit	가 사용 비용을 계산하는 데 AWS 사용한 요금 단위입니다. 예를 들어 Amazon EC2 인스턴스 사용량의 요금 단위는 시간입니다.

열 이름	설명
processor	<p>Amazon EC2 인스턴스의 프로세서를 설명합니다.</p> <p>예: High Frequency Intel Xeon E7-8880 v3(Haswell), Intel Xeon E5-2670, AMD EPYC 7571</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon DocumentDB • Amazon EC2 • Amazon Neptune • Amazon RDS • AWS 데이터베이스 마이그레이션 서비스
processor_features	<p>인스턴스의 프로세서 기능을 설명합니다.</p> <p>예: Intel AVX, Intel AVX2, Intel AVX512, Intel Turbo</p> <p>서비스:</p> <ul style="list-style-type: none"> • AWS 데이터베이스 마이그레이션 서비스 • Amazon DocumentDB • Amazon EC2 • Amazon Neptune • Amazon RDS

열 이름	설명
product_code	측정된 제품의 코드입니다. 예를 들어 AmazonEC2는 Amazon Elastic Compute Cloud의 제품 코드입니다.
product_family	제품 유형의 범주. 예: 경보, AWS 예산, 중지된 인스턴스, 스토리지 스냅샷, 컴퓨팅
product_from_location	사용량이 발생한 위치를 설명합니다. 샘플 값: 외부, 미국 동부(버지니아 북부), 글로벌 서비스: <ul style="list-style-type: none"> • Amazon CloudFront • AWS DataTransfer
product_group	정의상 유사하거나 함께 그룹화된 여러 제품의 구조입니다. 예를 들어 Amazon EC2 팀은 제품을 공유 인스턴스, 전용 호스트, 전용 사용으로 분류할 수 있습니다. 서비스: <ul style="list-style-type: none"> • AWS Certificate Manager • AWS CodeCommit • AWS Glue • AWS IoT 분석 • AWS Lambda

열 이름	설명
product_name	<p>AWS 서비스의 전체 이름입니다.</p> <p>예: AWS Backup, AWS Config, Amazon Registrar, Amazon Elastic File System, Amazon Elastic Compute Cloud</p>
product_to_location	<p>위치 사용 대상을 설명합니다.</p> <p>샘플 값: 외부, 미국 동부(버지니아 북부)</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon CloudFront • AWS DataTransfer
public_cost	<p>퍼블릭 온디맨드 인스턴스 요금 기반의 항목의 총 비용. 여러 온디맨드 퍼블릭 요금이 있는 SKU를 보유하고 있는 경우 가장 높은 티어에 상응하는 요금이 표시됩니다. 예를 들어 프리 티어 또는 계층화된 요금을 제공하는 서비스가 있습니다.</p>
purchase_option	<p>이 항목의 AWS 리소스에 대해 획득하고 비용을 지불한 방식입니다. purchase_option 열에는 절감형 플랜, 예약형 인스턴스 및 스팟 인스턴스에 대해 각각 'SavingsPlan', 'Reserved', 'Spot'이 포함됩니다. purchase_option 열에는 나머지 레코드에 대한 'OnDemand'가 있습니다.</p>

열 이름	설명
region	<p>AWS 서비스를 호스팅하는 지리적 영역입니다. 이 필드를 사용하여 특정 리전에 걸쳐 사용량을 분석합니다.</p> <p>예: eu-west-3, us-west-1, us-east-1, ap-northeast-2, sa-east-1</p> <p>서비스:</p> <ul style="list-style-type: none"> • Amazon EC2 • AWS Certificate Manager • Amazon S3 • Amazon RDS • Amazon DynamoDB • 전체 서비스 목록을 보려면 Column_Attribute_Service.zip을 다운로드하세요.
ri_sp_arn	<p>고유한 절감형 플랜 또는 예약형 인스턴스 식별자입니다. 일반적으로 <code>arn: aws: savingsplans: <region>:<account-id>:savingsplan/<savings-plan-id></code> 또는 <code>arn: aws:ec2:<region>:<account-id>:reserved-instances/<reserved-instance-id></code> 형식을 따릅니다.</p>

열 이름	설명
ri_sp_trueup	<p>이 값은 품목 유형에 따른 조정액 합계입니다. 트루업은 혼합되지 않은 비용을 사용한 기간에 발생한 총 선결제 요금과 상각된 비용을 사용하는 기간에 적용되는 선결제 요금 중 적은 금액의 차이를 나타냅니다. 조정액은 다음과 같이 결정합니다.</p> <ul style="list-style-type: none"> • 품목 유형이 'SavingsPlanRecurringFee'인 경우, 조정액은 절감형 플랜의 청구 기간에서 분할된 선결제 약정액을 뺀 것입니다. • 품목 유형이 'RIFee'인 경우, 조정액은 예약의 청구 기간에 대한 상각된 선결제 요금을 뺀 것입니다. • 다른 모든 품목 유형의 경우 조정액은 0입니다.
ri_sp_upfront_fee	<p>선결제 요금은 특정 유형의 예약형 인스턴스 또는 절감형 플랜을 선택할 때 지불하는 초기 비용을 말합니다.</p>
service	<p>AWS 서비스의 이름입니다.</p> <p>예: AmazonVPC, AmazonRDS, AmazonRoute53 등</p>

열 이름	설명
tenancy	<p>Amazon EC2 인스턴스에 허용되는 테넌시 유형입니다.</p> <p>예: 전용, 예약형, 공유, NA, 호스트</p> <p>서비스:</p> <ul style="list-style-type: none"> Amazon EC2 Amazon ECS
unblended_cost	UnblendedCost는 UnblendedRate에 UsageAmount를 곱한 값입니다.
usage_date	<p>이 품목의 시작 날짜와 시간(UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ 입니다.</p> <p>예: 2023-10-01T00:00:00.000Z</p>
usage_quantity	<p>지정된 기간 동안 발생한 사용량입니다. 크기가 유연한 예약 인스턴스의 경우 reservation/TotalReservedUnits 열을 대신 사용합니다.</p> <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note 특정 구독 요금의 UsageAmount는 0입니다.</p> </div>
usage_type	<p>이 항목의 사용량 세부 정보입니다. 예를 들어 USW2-BoxUsage:m2.2xlarge는 미국 서부(오레곤) 리전에 있는 M2 고용량 메모리 더블 엑스트라 라지 인스턴스를 설명합니다.</p>

탄소 배출량

탄소 배출량 테이블에는 예상 탄소 배출량이 포함되어 있습니다. 탄소 배출량 데이터에 대한 계정 수준 및 리전별 세부 정보를 제공합니다. 매월 자동으로 Amazon S3에 CSV 또는 Parquet 형식으로 데이터를 전송하도록 설정할 수 있어, 기존 비즈니스 인텔리전스 도구 및 보고 시스템과의 통합이 간편합니다. 방법론에 대한 자세한 내용은 AWS 지속 가능성 사용 설명서의 [계산 방법론](#)을 참조하세요.

탄소 배출량에 대한 SQL 테이블 이름은 CARBON_EMISSIONS입니다.

기록 데이터

설정 완료 후 24시간 이내에 2022년 1월까지의 데이터를 수신하게 되어, 수동 데이터 수집 없이도 기존 분석 및 추세 보고를 수행할 수 있습니다.

테이블 구성

탄소 배출량 테이블에 대한 테이블 구성은 없습니다.

권한

탄소 발자국 데이터에 액세스하려면 IAM 권한이 필요합니다. `sustainability:GetCarbonFootprintSummary`.

모델 버전

탄소 배출량 계산 방법은 시간이 지남에 따라 사용 패턴을 더 정확히 반영하고 탄소 회계 모범 사례에 부합하도록 발전할 것입니다. 내보내기는 'model_version=Y/' 및 'usage_period=YYYY-MM/'로 계층적 순서에 따라 분할됩니다. 내보내기가 저장되는 'model_version' 파티션은 해당 내보내기를 생성하는 데 사용된 모델 버전을 나타내며, 'usage_period' 파티션은 탄소 배출량이 생성된 날짜를 나타냅니다. 이 구조를 통해 파티션 이름을 확인함으로써 기존 모델과 새 모델로 생성된 데이터를 구분할 수 있습니다.

AWS 조직 지원

탄소 배출량 테이블은 AWS Organizations의 통합 결제 기능에서 설정한 설정을 상속합니다. 통합 결제가 활성화되면 관리 계정과 멤버 계정에 대한 다양한 동작이 존재합니다. 관리 계정을 사용하는 경우 탄소 배출량 테이블에는 관리 계정 및 조직의 모든 멤버 계정에 대한 예상 탄소 배출량 데이터가 포함됩니다. 멤버 계정을 사용하는 경우 탄소 배출량 테이블에는 해당 멤버 계정에 대한 예상 탄소 배출량 데이터만 포함됩니다.

멤버 계정이 새 조직에 가입하거나 관리 계정이 멤버 계정으로 전환되어 새 조직에 가입한 후에는 해당 계정의 탄소 배출량 데이터가 새 조직의 내보내기 기록에 포함됩니다. 각 관리 계정에는 해당 관리 계정에 연결된 기간 동안의 멤버 계정의 데이터가 포함됩니다. 예를 들어 멤버 계정이 월 15일에 조직 A를 떠나 조직 B에 가입합니다. 그런 다음 멤버 계정이 보고서를 생성합니다. 멤버 계정은 조직 B에 가입한 후 내보내기를 생성했으므로 멤버 계정의 해당 월 탄소 배출량 테이블 내보내기에는 해당 계정이 조직 B의 멤버였던 시점의 예상 탄소 배출량 데이터가 포함됩니다.

멤버 계정이 조직을 탈퇴하거나 독립 실행형 계정으로 전환하더라도, 멤버 계정은 해당 내보내기 파일이 저장된 Amazon S3 버킷에 대한 권한이 있는 경우 이전 내보내기 파일에 계속 액세스할 수 있습니다. 해지되거나 일시 중지된 계정과 관련된 탄소 배출량은 해당 계정이 활성화된 기간 동안 관리 계정 데이터 내보내기 파일에 표시됩니다.

자세한 내용은 [결제 사용 설명서의 AWS 조직에 대한 통합 결제](#)를 참조하세요. AWS

탄소 배출량 열

열 이름	설명	데이터 유형	Null 값 허용됨
last_refresh_timestamp	탄소 발자국 값이 마지막으로 생성된 시간 (UTC 기준)입니다. 형식은 YYYY-MM-DD HH:mm:ssZ 입니다.	timestamp	아니요
location	사용량의 오리지न 지역을 사람이 읽을 수 있는 위치로 설명하며, region_code에 해당합니다. 예를 들어, 미국 동부(버지니아 북부)입니다. Amazon CloudFront와 같은 글로벌 서비스의 배출량은 Global 항목으로 보고됩니다.	string	예
model_version	탄소 발자국 데이터가 계산된 방법론의 버	string	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
	전입니다. 예를 들어 v2.0.0입니다.		
payer_account_id	지급 계정의 계정 ID입니다. AWS Organizations에 속한 조직의 경우 이 항목은 관리 계정의 계정 ID입니다.	string	아니요
product_code	측정된 제품의 코드입니다. 예를 들어 Amazon EC2는 Amazon Elastic Compute Cloud의 제품 코드입니다.	string	예
region_code	리전은 데이터 센터가 클러스터링되는 전 세계의 물리적 위치입니다.는 논리적 데이터 센터의 각 그룹을 가용 영역(AZ)으로 AWS 호출합니다. 각 AWS 리전은 지리적 영역 내에서 물리적으로 분리된 여러 개의 격리된 AZs로 구성됩니다. 리전 코드 속성은 AWS 리전과 이름이 동일하며 AWS 서비스를 사용할 수 있는 위치를 지정합니다. 예를 들어 us-east-1입니다.	string	예

열 이름	설명	데이터 유형	Null 값 허용됨
total_lbm_emissions_unit	위치 기반 방법(LBM) 배출량에 사용되는 측정 단위입니다. 현재 지원하는 단위는 이산화탄소 등가물(MTCO2e)의 미터톤입니다.	string	예
total_lbm_emissions_value	해당 계정과 관련된 위치 기반 방법(LBM) 배출량 총 추정값입니다.	double	예
total_mbm_emissions_unit	시장 기반 방법(MBM) 배출량에 사용되는 측정 단위입니다. 현재 지원하는 단위는 이산화탄소 등가물(MTCO2e)의 미터톤입니다.	string	예
total_mbm_emissions_value	해당 계정과 관련된 시장 기반 방법(MBM) 배출량 총 추정값입니다.	double	예
total_scope_1_emissions_value	해당 계정과 관련된 범위 1 배출량 값입니다.	double	아니요
total_scope_1_emissions_unit	범위 1 배출량에 사용되는 측정 단위입니다.	string	아니요
total_scope_2_lbm_emissions_value	해당 계정과 연관된 범위 2 위치 기반 방법(LBM) 배출량 값입니다.	double	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
total_scope_2_lbm_emissions_unit	범위 2 위치 기반 방법 (LBM) 배출량에 사용되는 측정 단위입니다.	string	아니요
total_scope_2_mbm_emissions_value	해당 계정과 연관된 범위 2 시장 기반 방법 (MBM) 배출량 값입니다.	double	아니요
total_scope_2_mbm_emissions_unit	범위 2 시장 기반 방법 (MBM) 배출량에 사용되는 측정 단위입니다.	string	아니요
total_scope_3_lbm_emissions_value	해당 계정과 연관된 범위 3 위치 기반 방법 (LBM) 배출량 값입니다.	double	아니요
total_scope_3_lbm_emissions_unit	범위 3 위치 기반 방법 (LBM) 배출량에 사용되는 측정 단위입니다.	string	아니요
total_scope_3_mbm_emissions_value	해당 계정과 연관된 범위 3 시장 기반 방법 (MBM) 배출량 값입니다.	double	아니요
total_scope_3_mbm_emissions_unit	범위 3 시장 기반 방법 (MBM) 배출량에 사용되는 측정 단위입니다.	string	아니요

열 이름	설명	데이터 유형	Null 값 허용됨
usage_account_id	탄소 발자국 값과 연결된 계정의 계정 ID입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다.	string	아니요
usage_period_end	이 보고서의 기간 종료일(UTC)입니다. 형식은 YYYY-MM-DD HH:mm:ssZ 입니다.	timestamp	아니요
usage_period_start	이 보고서의 기간 시작일(UTC)입니다. 형식은 YYYY-MM-DD HH:mm:ssZ 입니다.	timestamp	아니요

데이터 내보내기 처리

다음 섹션에서는 데이터 내보내기 처리에 대한 정보를 찾을 수 있습니다.

Amazon Athena 구성

데이터 내보내기는 비용 및 사용량 보고서(CUR)와는 달리, 내보내기를 쿼리하도록 Athena를 설정하는 SQL 파일을 제공하지 않습니다. 데이터 내보내기에 CloudFormation 템플릿을 사용하거나(옵션 1 참조) Athena를 수동으로 구성해야 합니다(옵션 2 참조).

(선택 사항 1) CloudFormation 템플릿 사용: CloudFormation 템플릿과 데이터 내보내기를 사용하여 Athena를 설정하는 지침을 찾으려면 [Cloud Intelligence Dashboards 프레임워크의 데이터 내보내기를](#) 참조하세요.

(옵션 2) AWS Glue 크롤러를 사용하여 Athena용 테이블 및 파티션 구축: Athena에 대한 CUR 또는 탄소 배출량 데이터 내보내기를 생성할 때 Apache Parquet 파일 형식을 사용하는 것이 좋습니다. 더 작고 저렴한 Athena 쿼리에 기여하는 더 나은 압축 및 열 지향 스토리지를 제공합니다. 각 월별 파티션에 항상 각 파일의 복사본이 하나만 포함되고 Amazon Athena를 사용하여 쿼리를 실행할 때 중복 항목이 표시되지 않도록 전송 기본 설정 덮어쓰기가 필요합니다.

또한 AWS Glue 크롤러와 함께 Glue를 사용하여 Athena에 데이터를 로드하는 것이 좋습니다.

Glue AWS 크롤러를 사용하여 Athena용 테이블 및 파티션을 빌드하려면

1. 다음 데이터 내보내기 전송 옵션을 사용하여 CUR 2.0 또는 탄소 배출량 내보내기를 생성합니다.
 - 압축 유형 및 파일 형식: Parquet - Parquet
 - 파일 버전 관리: 기존 데이터 내보내기 파일 덮어쓰기
2. Athena에서 Trino SQL과 함께 노트북 편집기를 사용하고 생성을 선택하여 "AWS Glue 크롤러"로 테이블을 생성합니다. Glue 크롤러 워크플로를 사용하여 Glue 크롤러가 s3://<bucket-name>/<prefix>/<export-name>/data 폴더에서 실행되도록 하여 Athena로 지정된 내보내기를 위해 전달된 모든 파티션을 자동으로 로드시킵니다.
3. Glue 크롤러가 완료되면 Athena를 사용하여 Glue 크롤러에서 생성한 테이블에 쿼리를 작성할 수 있습니다.

Amazon Redshift 구성

Amazon Redshift는 프로비저닝된 용량 또는 서버리스 모델로 액세스할 수 있는 클라우드 데이터 웨어하우스입니다. Amazon Redshift는 데이터 내보내기에서 데이터를 처리하기 위한 빠른 쿼리 성능을 제공합니다.

현재 데이터 내보내기는 비용 및 사용량 보고서(CUR)와는 달리, 내보내기를 쿼리하도록 Redshift를 설정하는 데 필요한 SQL 파일을 제공하지 않습니다. 하지만 Redshift를 수동으로 설정하여 내보내기를 쿼리할 수는 있습니다. Redshift에는 gzip/csv 압축 및 파일 형식을 사용하는 것이 좋습니다.

Amazon Redshift 설정에 관한 자세한 내용은 [Amazon Redshift 시작 안내서](#)를 참조하세요.

CUR 2.0 처리를 위한 권장 SQL 쿼리

CUR 2.0 내보내기 데이터를 Amazon Athena 또는 Amazon Redshift와 같은 데이터 분석 도구에 로드한 후 이를 처리하여 비용 및 사용량 인사이트를 얻을 수 있습니다. AWS Well-Architected Labs는 CUR을 처리하는 데 사용할 수 있는 CUR 쿼리 라이브러리를 제공합니다. 자세한 내용은 [AWS CUR 쿼리 라이브러리](#)를 참조하세요.

SQL 쿼리에 대한 다음 두 가지 정보를 참고하세요.

- Well-Architected Labs SQL 쿼리는 데이터 내보내기 쿼리 필드에서 작동하지 않습니다. 데이터 내보내기는 집계 및 이러한 쿼리에 사용되는 일부 다른 SQL 구문을 지원하지 않기 때문입니다.
- Well-Architected Labs SQL 쿼리는 열 이름을 기본 이름에서 변경하지 않은 경우에만 작동합니다. 쿼리에 따라 점 연산자를 사용하여 일부 제품 열을 별도의 열로 쿼리해야 할 수도 있습니다. 자세한 내용은 [데이터 쿼리-SQL 쿼리 및 테이블 구성](#)을 참조하세요.

탄소 배출량 데이터 내보내기 처리를 위한 권장 SQL 쿼리

payer_account_id당 총 탄소 배출량을 구하는 방법:

```
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 및 product_code당 총 탄소 배출량을 구하는 방법:

```
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 및 region_code당 총 탄소 배출량을 구하는 방법:

```
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;
```

Cost and Usage Dashboard의 이해

Cost and Usage Dashboard는 [Cloud Intelligence Dashboards](#)(CID) 오픈 소스 프로젝트에서 영감을 얻어 Amazon QuickSight에서 제공하는 배포하기 쉽고 안전하며 사전 구축된 대시보드입니다. Cost and Usage Dashboard에는 6개의 Cloud Intelligence Dashboards 중 하나인 [CUDOS 대시보드](#)에서 리소스 수준 뷰를 제외하고 요약 비주얼의 일부가 포함되어 있습니다. Cost and Usage Dashboard는 CUDOS 솔루션의 이점을 설정하기 쉬운 AWS 결제 및 비용 관리 콘솔 기능으로 통합하여 Amazon Athena 뷰 또는 AWS Glue 크롤러와 같은 기본 인프라를 유지 관리할 필요가 없도록 합니다. AWS 결제 및 비용 관리 콘솔의 데이터 내보내기 페이지에서 몇 분 만에 Cost and Usage Dashboard를 배포할 수 있습니다. CID에는 AWS CloudFormation 템플릿 기반 배포가 포함됩니다. 전체 CID 솔루션 설정에 대한 자세한 내용은 [AWS Well-Architected Labs](#)를 참조하세요.

다음 테이블에는 Cost and Usage Dashboard와 Cloud Intelligence Dashboards(CID)의 차이점이 요약되어 있습니다.

Feature	Cost and Usage Dashboard	Cloud Intelligence Dashboards
배포	콘솔을 AWS 통한 원활한 배포	CloudFormation, Command Line 또는 Terraform
AWS Organizations를 위한 배포 옵션	관리 계정에서만	관리 계정 또는 위임된 연결 계정에서
여러 AWS Organizations 취합	아니요	예
개괄적인 수준의 비용 및 사용량 인사이트	예	예
리소스 수준의 세부 정보	아니요	예
예약형 인스턴스 및 절감형 플랜 인사이트	아니요	예
지원되는 데이터 원본	비용 및 사용량 요약(대시보드 뷰)	비용 및 사용량 보고서(CUR), Compute Optimizer, Trusted Advisor, 비용 이상 탐지

비용 및 사용량 보고서(CUR)의 이해

CUR 및 CUR 2.0을 사용하여 AWS 비용 및 사용량에 대한 가장 자세한 정보를 얻을 수 있습니다. 절감형 플랜 및 예약형 인스턴스(RI) 사용률, 요금, 분할 비용 할당 추적과 같은 사용 사례에 대한 자세한 내용은 [Use cases](#)를 참조하세요.

탄소 배출량 데이터 내보내기 이해

탄소 배출량 데이터 내보내기를 사용하여 지금인 계정에 연결된 모든 사용 계정에 대한 탄소 배출량 추정치를 얻을 수 있습니다. 이러한 탄소 배출량을 계산하고 분류하는 방법에 대한 자세한 내용은 AWS 지속 가능성 사용 설명서의 [계산 방법을](#) 참조하세요.

보안 및 권한

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 보안에 가장 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

보안은 AWS 와 사용자 간의 공동 책임입니다. [공동 책임 모델](#)은 이를 클라우드의 보안과 클라우드 내 보안으로 설명합니다.

클라우드 보안: AWS 는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다. AWS 또한는 안전하게 사용할 수 있는 서비스를 제공합니다. 서드 파티 감사자는 정기적으로 [AWS 규정 준수 프로그램](#)의 일환으로 보안 효과를 테스트하고 검증합니다. AWS 비용 관리에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 [AWS 규정 준수 프로그램 제공 범위 내 서비스](#)를 참조하세요.

클라우드의 보안: 사용자의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다. 이 설명서는 Billing and Cost Management 사용 시 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목적에 맞게 Billing and Cost Management를 구성하는 방법을 보여줍니다. 또한 Billing and Cost Management 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

주제

- [데이터 내보내기에 대한 ID 및 액세스 관리](#)
- [데이터 내보내기의 데이터 보호](#)

데이터 내보내기에 대한 ID 및 액세스 관리

AWS Identity and Access Management(IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어할 수 있도록 지원하는 AWS 서비스입니다. IAM 관리자는 어떤 사용자가 결제 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 받을 수 있는지 제어합니다. IAM은 추가 비용 없이 사용할 수 있는 AWS 서비스입니다.

데이터 내보내기를 사용하려면 IAM 사용자에게 IAM 내 bcm-data-exports namespace에서의 작업에 대한 액세스 권한을 부여해야 합니다. 사용 가능한 작업은 아래 테이블을 참조하세요.

데이터 내보내기 작업	설명	액세스 레벨	조건 키	조건 키
CreateExport	사용자가 내보내기를 생성하고 쿼리, 전송 구성, 일정 구성 및 콘텐츠 구성을 지정하도록 허용합니다.	쓰기	내보내기 테이블	aws:RequestTag/\${TagKey} aws:TagKeys
UpdateExport	사용자가 기존 내보내기를 업데이트하도록 허용합니다.	쓰기	내보내기 테이블	aws:ResourceTag/\${TagKey}
DeleteExport	사용자가 기존 내보내기를 삭제하도록 허용합니다.	쓰기	내보내기	aws:ResourceTag/\${TagKey}
GetExport	사용자가 기존 내보내기를 보도록 허용합니다.	읽기	내보내기	aws:ResourceTag/\${TagKey}
ListExports	사용자가 기존의 모든 내보내기를 나열하도록 허용합니다.	읽기		
GetExecution	사용자가 내보낸 데이터의 메타데이터 및 스키마를 포함하여 지정된 실행의 세부 정보를 보도록 허용합니다.	읽기	내보내기	aws:ResourceTag/\${TagKey}

데이터 내보내기 작업	설명	액세스 레벨	조건 키	조건 키
ListExecutions	사용자가 제공된 내보내기 식별자의 모든 실행을 나열하도록 허용합니다.	읽기	내보내기	aws:ResourceTag/\${TagKey}
GetTable	사용자가 지정된 테이블의 스키마를 가져오도록 허용합니다.	읽기	테이블	
ListTables	사용자가 사용할 수 있는 모든 테이블을 나열하도록 허용합니다.	읽기		
TagResource	사용자가 기존 내보내기에 태그를 지정하도록 허용합니다.	쓰기	내보내기	aws:ResourceTag/\${TagKey} aws:RequestTag/\${TagKey} aws:TagKeys
UntagResource	사용자가 기존 내보내기의 태그 지정을 취소하도록 허용합니다.	쓰기	내보내기	aws:ResourceTag/\${TagKey} aws:TagKeys
ListTagsForResource	사용자가 기존 내보내기와 연결된 태그를 나열하도록 허용합니다.	읽기	내보내기	aws:ResourceTag/\${TagKey}

이러한 컨텍스트 키를 사용하는 방법에 대한 자세한 내용은 IAM 사용 설명서에서 [태그를 사용한 AWS 리소스 액세스 제어](#) 섹션을 참조하세요.

다음 테이블에서는 데이터 내보내기에서 사용할 수 있는 리소스 유형을 설명합니다.

리소스 유형	설명	ARN
내보내기	내보내기는 CreateExport API로 생성되는 리소스입니다. 내보내기는 결제 및 비용 관리 쿼리 출력이 반복적으로 생성됩니다.	arn:\${Partition}:bcm-data-exports:\${Region}:\${Account}:export/\${exportName}-{UUID}
테이블	테이블은 사용자가 내보내기를 통해 쿼리하는 행-열 형식의 데이터입니다. 테이블은 고객을 AWS 위해에서 생성하고 관리합니다. 테이블은 고객이 삭제할 수 없습니다.	arn:\${Partition}:bcm-data-exports:\${Region}:\${Account}:table/\${TableName}

데이터 내보내기에서 COST_AND_USAGE_REPORT 또는 COST_AND_USAGE_DASHBOARD 테이블 리소스의 내보내기를 생성하려면 IAM 사용자에게 IAM 내 각 cur 작업에 대한 권한도 있어야 합니다. 즉, cur에 대한 명시적 허용이 없거나 서비스 제어 정책(SCP)에서 cur에 대해 명시적 거부를 제공하는 등 어떤 이유로든 IAM 사용자가 cur 작업을 사용하지 못하도록 차단되면 해당 IAM 사용자는 이 두 테이블의 내보내기를 생성하거나 업데이트할 수 없습니다.

다음 테이블은 이 두 테이블에 대해 데이터 내보내기에서 어떤 bcm-data-exports 작업에 어떤 cur 작업이 필요한지 보여줍니다.

데이터 내보내기 작업	테이블 리소스	IAM의 추가 필수 작업
bcm-data-exports:CreateExport	COST_AND_USAGE_REPORT	cur:PutReportDefinition
	COST_AND_USAGE_DASHBOARD	

샘플 정책

IAM 사용자가 데이터 내보내기에서 CUR 2.0 내보내기에 대한 전체 액세스 권한을 갖도록 허용합니다.

JSON

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

결제 및 비용 관리에서 데이터 내보내기를 사용하기 위한 액세스 제어와 IAM 권한에 대한 자세한 내용은 [액세스 권한 관리 개요](#)를 참조하세요.

견적 AWS CUR 2.0 생성

견적 CUR 2.0을 생성하려면 다음 IAM 정책을 포함해야 합니다.

IAM 사용자가 CUR 2.0 및 결제 그룹 결제 보기에 대한 전체 액세스 권한을 갖도록 허용합니다.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCreateCur20AnyBillingView",
      "Effect": "Allow",
      "Action": "bcm-data-exports:CreateExport",
      "Resource": [
        "arn:aws:bcm-data-exports:*:*:table/COST_AND_USAGE_REPORT",
        "arn:aws:bcm-data-exports:*:*:export/*",
        "arn:aws:billing:*:*:billingview/*"
      ]
    }, {
      "Sid": "CurDataAccess",
      "Effect": "Allow",
      "Action": "cur:PutReportDefinition",
      "Resource": "*"
    }
  ]
}
```

IAM 역할이 특정 결제 그룹에 액세스할 수 있도록 하려면 역할이 액세스할 수 있는 결제 보기 ARN을 추가할 수 있습니다.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCreateSpecificBillingViewCur20",
      "Effect": "Allow",
```

```

    "Action": "bcm-data-exports:CreateExport",
    "Resource": [
      "arn:aws:bcm-data-exports:*:*:table/COST_AND_USAGE_REPORT",
      "arn:aws:bcm-data-exports:*:*:export/*",
      "arn:aws:billing:444455556666:billingview/billing-
group-111122223333"
    ]
  }, {
    "Sid": "CurDataAccess",
    "Effect": "Allow",
    "Action": "cur:PutReportDefinition",
    "Resource": "*"
  }
]
}

```

데이터 내보내기의 데이터 보호

Data Exports에서 AWS 공동 책임 모델을 데이터 보호에 적용하는 방법을 알아봅니다.

S3 보안 모범 사례

데이터 내보내기는 결제 및 비용 관리 데이터를 Amazon S3 버킷으로 전송합니다. S3 버킷의 보안을 보장하기 위해 취할 수 있는 여러 단계가 있습니다. 자세한 내용은 Amazon S3 사용 설명서의 [Amazon S3 보안 모범 사례](#)를 참조하세요.

S3에서 데이터 암호화

기본적으로, 데이터 내보내기는 Amazon S3 관리형 암호화 키를 통한 서버 측 암호화(SSE-S3)를 사용하여 암호화됩니다. Amazon Key Management Service(KMS) 암호화(SSE-KMS)를 사용하여 내보내기를 암호화하려면 내보내기가 전송된 후 KMS로 암호화를 트리거해야 합니다. 자세한 내용은 Amazon S3 사용 설명서의 [Amazon S3 버킷에 대한 기본 서버 측 암호화 동작 설정](#)을 참조하세요.

할당량 및 제한

다음 테이블에서는 데이터 내보내기의 현재 할당량 및 제한 사항에 대해 설명합니다.

할당량 유형	할당량 값
CUR 2.0 (COST_AND_USAGE_REPORT) 의 내보내기 개수	5
비용 최적화 권장 사항 테이블의 내보내기 수(COST_OPTIMIZATION_RECOMMENDATIONS)	2
FOCUS 1.2 테이블의 내보내기 수(FOCUS_1_2_AWS)	2
FOCUS 1.0 테이블의 내보내기 수(FOCUS_1_0_AWS)	2
Cost and Usage Dashboard 테이블 (COST_AND_USAGE_DASHBOARD)의 내보내기 수	2
탄소 배출량 테이블의 내보내기 수(CARBON_EMISSIONS)	2
결제 이체 계정의 레거시 내보내기 수	1000

문제 해결

데이터 내보내기 또는 비용 및 사용 보고서(CUR)를 사용하면 문제 해결이 필요한 특정 문제 또는 문제가 발생할 수 있습니다. 이 문제 해결 섹션에서 일반적인 문제를 빠르고 효율적으로 해결하는 데 도움이 되는 권장 사항을 참조하세요.

주제

- [일반 문제 해결](#)
- [CUR 2.0 문제 해결](#)
- [Cost and Usage Dashboard 문제 해결](#)
- [비용 및 사용 보고서 문제 해결](#)
- [탄소 배출량 데이터 내보내기 문제 해결](#)

일반 문제 해결

주제

- [내보내기 상태가 비정상인 이유는 무엇인가요?](#)
- [Data Exports에서 내 SQL 문을 수락하지 않는 이유는 무엇인가요?](#)
- [데이터 내보내기 내에서 Athena를 구성하기 위해 미리 정의된 SQL 스크립트를 찾을 수 없습니다. 그 이유는 무엇인가요?](#)
- [내보내기 파티션 중 하나가 비어 있는 이유는 무엇인가요?](#)
- [Amazon S3 버킷에 보고서 파일이 없는 이유는 무엇인가요?](#)

내보내기 상태가 비정상인 이유는 무엇인가요?

'비정상'인 내보내기는 Amazon S3 버킷에 새로 고침을 마지막으로 전송하려고 시도했을 때 오류가 발생한 것을 말합니다. '비정상' 메시지 위에 커서를 올리거나 GetExport API를 호출하면 다음 오류 메시지 중 하나가 표시될 수 있습니다.

- Data Exports 문제
 - 권한이 충분하지 않음: Data Exports가 내보내기 파일을 S3 버킷으로 전송할 수 없었음을 의미합니다. 이 문제는 [Amazon S3 버킷에 데이터 내보내기 설정](#)에 나열된 권한으로 S3 버킷 정책을 업데이트하여 해결할 수 있습니다.

- 청구서 소유자 변경됨: 이 오류는 AWS 계정이 새 조직으로 이동하거나 AWS Organizations에 조직을 떠날 때 발생할 수 있습니다. 조직에 속해 있고 AWS Billing Conductor의 결제 그룹에 속해 있는지 여부에 관계없이 관리 계정이 변경될 때도 발생할 수 있습니다. 이 문제를 해결하는 가장 좋은 방법은 새 CUR을 만들고 기존 CUR을 삭제하는 것입니다. 계정의 조직 또는 청구 그룹이 변경되지 않았어야 한다고 생각되면 계정 관리자에게 문의하세요.
- 내부 장애: 이 오류는 Data Exports 내부 인프라 문제로 인해 발생합니다. AWS 서비스 상태 대시보드에서 데이터 내보내기에 영향을 미칠 수 있는 서비스 전반의 문제에 대한 업데이트를 검토하거나 AWS Support에 문의하여 자세한 내용이나 도움을 받으세요.
- QuickSight 통합 문제
 - SPICE 용량 부족: 이 오류는 QuickSight에 비용 및 사용량 데이터를 모으기에 충분한 처리 용량이 프로비저닝되지 않았음을 의미합니다. SPICE 용량을 늘리는 방법에 대한 자세한 내용은 [SPICE 메모리 용량 관리](#)를 참조하세요.
 - 매니페스트 파일에 액세스할 권한이 충분하지 않음: S3 버킷에 액세스하기 위해 QuickSight에 할당된 서비스 역할이 더 이상 작동하지 않습니다. 서비스 정책을 검토하여 비용 및 사용량 데이터를 저장하는 S3 버킷에 대한 읽기 권한을 부여하고 있는지 확인하세요.
 - 매니페스트 파일 액세스 시도 시 액세스 거부됨: 내보내기 파일을 저장하는 S3 버킷에 액세스하여 이 내보내기에 대한 QuickSight 대시보드가 있는지 확인할 수 있는 액세스 권한이 IAM 역할에 없습니다. 대시보드가 작동하고 있을 수도 있고 작동하지 않고 있을 수도 있습니다. QuickSight 대시보드를 확인하려면 내보내기 데이터를 저장하는 S3 버킷에 대한 s3:GetObject 권한이 필요합니다.
 - QuickSight CreateBundle 실패: 이 오류는 QuickSight에서 대시보드를 만들지 못했음을 의미합니다. 이 문제는 새 서비스 역할을 생성했거나 적절한 권한이 없는 기존 서비스 역할을 선택한 경우 IAM 역할 전파가 지연되었기 때문에 발생할 수 있습니다. 새 서비스 역할을 생성한 경우 재시도 작업을 사용하세요. 기존 서비스 역할을 선택한 경우 내보내기를 삭제하고 새 서비스 역할을 사용하여 내보내기를 새로 만들어야 합니다.
 - 대시보드가 존재하지 않음: 이 오류는 QuickSight에서 대시보드가 삭제되었음을 의미합니다. 데이터 내보내기에서 기존 Cost and Usage Dashboard 내보내기를 삭제하고 다시 생성해야 합니다.
 - QuickSight 계정이 존재하지 않음: 이 오류는 QuickSight 계정이 삭제되었음을 의미합니다. 대시보드를 다시 사용하려면 QuickSight 계정을 다시 만들어야 합니다. QuickSight 계정을 다시 만든 후에는 데이터 내보내기에서 기존 Cost and Usage Dashboard 내보내기를 삭제하고 다시 생성해야 합니다.

Data Exports에서 내 SQL 문을 수락하지 않는 이유는 무엇인가요?

Data Exports는 주로 열 선택 및 행 필터에 초점을 맞춘 제한된 SQL 구문 세트를 지원합니다. SQL 문에는 관련 키워드와 연산자만 사용해야 합니다. 자세한 내용은 [데이터 쿼리](#)를 참조하세요.

데이터 내보내기 내에서 Athena를 구성하기 위해 미리 정의된 SQL 스크립트를 찾을 수 없습니다. 그 이유는 무엇인가요?

데이터 내보내기는 비용 및 사용량 보고서(CUR)와는 달리, 내보내기를 쿼리하도록 Athena를 설정하는 SQL 파일을 제공하지 않습니다. 데이터 내보내기에 CloudFormation 템플릿을 사용하거나 Athena를 수동으로 구성해야 합니다. 자세한 내용은 [Amazon Athena 구성](#)을 참조하세요.

내보내기 파티션 중 하나가 비어 있는 이유는 무엇인가요?

내보내기가 대부분의 애플리케이션에서 처리할 수 있는 것보다 큰 경우는 보고서를 여러 파일로 AWS 분할합니다. 내보내기 업데이트가 이전 내보내기보다 작고 “덮어쓰기” 모드를 사용하는 경우는 불필요한 파티션을 빈 데이터로 AWS 덮어씁니다. 내보내기 매니페스트는 데이터가 있는 파티션만 나열합니다. 보고서의 매니페스트 파일을 검토하여 수집할 필요가 없는 빈 파일을 찾습니다.

Amazon S3 버킷에 보고서 파일이 없는 이유는 무엇인가요?

Amazon S3 버킷 정책이 파일을 버킷에 넣을 수 있는 [billingreports.amazonaws.com](#) 서비스 권한을 부여하는지 확인합니다. 필요한 버킷 정책에 대한 자세한 내용은 [데이터 내보내기를 위한 Amazon S3 버킷 설정](#) 또는 [비용 및 사용 보고서를 위한 Amazon S3 버킷 설정](#)을 참조하세요.

CUR 2.0 문제 해결

주제

- [CUR에서 사용 가능했던 일부 열이 CUR 2.0에서는 표시되지 않는 이유는 무엇인가요?](#)
- [레거시 비용 및 사용 보고서는 어떻게 되나요? 사용이 중단되나요?](#)
- [CUR 2.0 내보내기를 생성하면 레거시 CUR에 영향을 미치나요?](#)
- [데이터 내보내기 및 CUR 테이블을 사용할 수 있는 IAM 권한이 있음에도 CUR 2.0 내보내기를 생성할 수 없는 이유는 무엇인가요?](#)
- [레거시 CUR 열과 동일한 CSV 형식으로 데이터 내보내기를 생성하려고 하면 "Invalid QueryStatement" 오류가 발생합니다. 이 문제를 해결하려면 어떻게 해야 하나요?](#)

- [데이터 내보내기 CUR 2.0으로 마이그레이션한 후 레거시 CUR 내보내기와 CUR 2.0 내보내기를 동시에 사용할 수 있나요?](#)
- [CUR 2.0의 내보내기를 생성하려고 할 때 "이 계정은 이 테이블에 대한 내보내기를 생성할 수 없습니다"라는 오류가 발생합니다. CUR 2.0 내보내기를 생성할 수 없는 이유는 무엇인가요?](#)

CUR에서 사용 가능했던 일부 열이 CUR 2.0에서는 표시되지 않는 이유는 무엇인가요?

CUR 2.0에서는 4개의 열 유형이 4개의 개별 열로 중첩되었습니다. 중첩된 열은 product, discount, resource_tag 및 cost_category입니다.

레거시 CUR에는 이름이 해당 문자열로 시작하는 수백 개의 열이 있을 수 있습니다. 변형은 AWS 서비스 또는 계약의 고객 사용에 따라 달라집니다. 이 스키마 설계로 인해 수백 개의 열이 드문드문 채워지는 경우가 발생했습니다. 열의 변동성으로 인한 충전 스키마가 원인이 되어 SQL 쿼리에 문제가 발생할 수도 있습니다.

따라서 열마다 다를 수 있는 AWS 열이 4개 열로 함께 중첩되었습니다. 일반적으로 사용되는 특정 제품 열은 중첩되지 않았습니다.

SQL의 점 연산자를 사용하여 CUR 2.0 내보내기에서 CUR 스키마를 다시 만들 수 있습니다. 이 작업을 수행하는 방법을 알아보려면 [Data Exports CUR에서 CUR 2.0으로 마이그레이션](#)을 참조하세요.

레거시 비용 및 사용 보고서는 어떻게 되나요? 사용이 중단되나요?

현재 레거시 CUR의 사용을 중단할 계획은 없습니다. 그러나 데이터 내보내기의 CUR 2.0은 일관된 스키마, 중첩된 데이터 및 추가 열(bill_payer_account_name 및 line_item_usage_account_name)과 같은 몇 가지 개선 사항을 제공하므로 CUR 2.0으로 마이그레이션하는 것이 좋습니다.

목표 날짜는 정해지지 않았지만 콘솔의 레거시 페이지에서 비용 및 사용 보고서를 최종적으로 사용 중단할 계획입니다. 그러나 레거시 CUR을 생성, 업데이트 및 삭제하는 것과 동일한 모든 기능은 데이터 내보내기 콘솔 페이지를 통해 사용할 수 있습니다.

Note

세부 결제 보고서(DBR)는 또 다른 기존 결제 특성이며, 추후 사용 중단될 수 있습니다. 해당 특성은 2019년 7월 8일부터 신규 사용자에게 제공되지 않고 있습니다.

CUR 2.0 내보내기를 생성하면 레거시 CUR에 영향을 미치나요?

CUR과 CUR 2.0은 서로 다른 두 가지 보고서입니다. CUR 2.0을 생성할 때 기존 CUR 설정에 영향을 주지 않습니다. 기본 설정에 따라 레거시 CUR과 CUR 2.0 중에서 선택할 수 있습니다.

데이터 내보내기 및 CUR 테이블을 사용할 수 있는 IAM 권한이 있음에도 CUR 2.0 내보내기를 생성할 수 없는 이유는 무엇인가요?

`cur:PutReportDefinition`에 대한 IAM 권한도 있어야 합니다.

레거시 CUR 열과 동일한 CSV 형식으로 데이터 내보내기를 생성하려고 하면 "Invalid QueryStatement" 오류가 발생합니다. 이 문제를 해결하려면 어떻게 해야 하나요?

현재 CSV 형식에서 레거시 CUR 열 이름을 맞추기 위해 “/” 같은 특수 문자를 포함해 열 이름을 변경할 수 없습니다. 지원되는 문자 유형에 대한 자세한 내용은 [SQL 쿼리](#)를 참조하세요.

데이터 내보내기 CUR 2.0으로 마이그레이션한 후 레거시 CUR 내보내기와 CUR 2.0 내보내기를 동시에 사용할 수 있나요?

예. 최대 10개의 레거시 CUR 내보내기와 5개의 CUR 2.0 내보내기를 동시에 사용할 수 있습니다.

CUR 2.0의 내보내기를 생성하려고 할 때 "이 계정은 이 테이블에 대한 내보내기를 생성할 수 없습니다"라는 오류가 발생합니다. CUR 2.0 내보내기를 생성할 수 없는 이유는 무엇인가요?

레거시 CUR과 달리 CUR 2.0은 현재 예상 결제 데이터로 CUR 2.0 내보내기를 생성하는 기능을 지원하지 않습니다. AWS Billing Conductor의 결제 그룹에 속한 경우 견적 결제 데이터만 수신할 수 있습니다. 따라서 CUR 2.0 내보내기를 생성하려고 할 때 이 오류 메시지가 표시됩니다. 여전히 레거시 CUR 내보내기를 생성할 수 있습니다.

Cost and Usage Dashboard 문제 해결

주제

- [Cost and Usage Dashboard를 만든 직후에 내보내기가 실패하는 이유는 무엇인가요?](#)
- [대시보드에 액세스할 수 없는 이유는 무엇인가요?](#)

- [대시보드를 보려고 할 때 QuickSight 계정 구독을 취소하라는 콘솔 관리 페이지로 이동하는 이유는 무엇인가요?](#)
- [방금 만든 Cost and Usage Dashboard에 이번 달의 데이터가 보이지 않는 이유는 무엇인가요?](#)
- [Cost and Usage Dashboard에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?](#)
- [QuickSight 대시보드 링크가 Data Exports 콘솔 페이지에서 사라진 이유는 무엇인가요?](#)
- [CUR 2.0에서 리소스 태그를 시각화하도록 Amazon QuickSight를 구성하려면 어떻게 해야 하나요?](#)

Cost and Usage Dashboard를 만든 직후에 내보내기가 실패하는 이유는 무엇인가요?

IAM 역할 전파 지연으로 인해 Cost and Usage Dashboard 내보내기가 실패했을 수 있습니다. 이 내보내기에 대한 새 서비스 역할을 생성한 경우 Amazon QuickSight는 S3 버킷에 액세스하여 대시보드를 생성할 권한이 없었을 수 있습니다. 내보내기 상태에서 '매니페스트 파일에 액세스할 수 있는 권한이 충분하지 않음' 오류가 표시되면 내보내기를 선택한 다음, 테이블 작업 메뉴에서 재시도를 선택하세요.

Cost and Usage Dashboard 내보내기를 위한 새 서비스 역할을 만들지 않은 경우 QuickSight에서 사용할 서비스 역할을 잘못 지정했을 수 있습니다. 이 경우 내보내기를 삭제하고 다시 만드는 동시에 Cost and Usage Dashboard 콘솔 워크플로에서 새 서비스 역할을 만들어야 합니다.

대시보드에 액세스할 수 없는 이유는 무엇인가요?

조회 권한이 없는 경우 Amazon QuickSight의 Cost and Usage Dashboard에 액세스하지 못할 수 있습니다. 문제를 해결하려면 내보내기 이름을 선택하여 내보내기를 여세요. QuickSight 생성자 필드를 확인하여 대시보드를 만든 사람을 확인하세요. 만든 사람에게 대시보드를 볼 수 있는 권한을 요청하세요.

대시보드를 보려고 할 때 QuickSight 계정 구독을 취소하라는 콘솔 관리 페이지로 이동하는 이유는 무엇인가요?

'Active Directory' 인증 방법을 사용하는 경우 이 오류가 발생할 수 있습니다. 내보내기의 세부 정보를 보려면 Cost and Usage Dashboard 내보내기 이름을 선택하세요. QUICKSIGHT 로그인을 선택하여 QuickSight 계정에 로그인하세요. 대시보드를 볼 수 있는 권한이 있는 경우 대시보드를 볼 수 있습니다.

방금 만든 Cost and Usage Dashboard에 이번 달의 데이터가 보이지 않는 이유는 무엇인가요?

모든 데이터가 대시보드에 입력되는 데 최대 24시간이 걸릴 수 있으므로 Cost and Usage Dashboard에 이번 달의 데이터가 누락되었을 수 있습니다. Cost and Usage Dashboard 내보내기 상태를 확인

하세요. 내보내기 상태가 '정상'으로 표시되는 경우 대시보드가 이번 달의 데이터로 업데이트될 때까지 24시간이 걸릴 수 있습니다. 24시간 후에도 대시보드에 이번 달 데이터가 표시되지 않으면 AWS Support에 문의하세요. 데이터 내보내기 콘솔 페이지의 내보내기 및 대시보드 테이블에서 Cost and Usage Dashboard 생성 시간을 확인할 수 있습니다.

Cost and Usage Dashboard에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?

Cost and Usage Dashboard에 다음과 같은 이유로 6개월간의 과거 데이터가 누락되었을 수 있습니다.

- 과거 데이터 없음: 새 계정이 되거나 최근에 AWS Organizations의 멤버십이 변경되어 6개월의 과거 지출이 없는 계정이 있는 경우 과거 데이터가 대시보드를 채울 수 없습니다.
- 과거 데이터 채우기가 아직 진행 중임: Data Exports를 통한 과거 데이터 채우기를 완료하는 데 최대 24시간이 걸릴 수 있습니다. SDK/CLI를 사용하여 이 내보내기에 대한 ListExecutions API에서 실패한 채우기 실행이 있는지 또는 아직 진행 중인지 확인할 수 있습니다. 조금 더 기다리거나 ListExecutions를 사용하여 채우기가 진행 중이 아닌지 확인하세요.
- 과거 데이터 채우기 실패: 내부 오류로 인해 과거 데이터 채우기가 완료되지 않았을 수 있습니다. 24시간이 지나도 채우기가 완료되지 않은 경우 실패했다는 결론을 내릴 수 있습니다. 또는 SDK/CLI의 ListExecutions API를 사용하여 이 내보내기에 실패한 실행이 있는지 찾아볼 수 있습니다. 채우기가 실패했다고 생각되면 콘솔에서 Cost and Usage Dashboard를 다시 만들어 보세요. 두 번째로 실패하는 경우 AWS Support에 문의하는 것이 좋습니다.

QuickSight 대시보드 링크가 Data Exports 콘솔 페이지에서 사라진 이유는 무엇인가요?

Data Exports 콘솔 페이지는 내보내기가 연결된 QuickSight 대시보드를 식별하기 위해 S3 버킷의 파일에서 읽습니다. 이 파일이 변경되거나 삭제되면 콘솔은 이 내보내기에 대한 대시보드가 존재하는지 알 수 없습니다. 대시보드가 QuickSight에 여전히 존재하지만 링크를 다시 표시하려면 이 파일을 복구해야 합니다.

CUR 2.0에서 리소스 태그를 시각화하도록 Amazon QuickSight를 구성하려면 어떻게 해야 하나요?

Cost and Usage Dashboard 기능은 리소스 태그 시각화를 지원하지 않습니다. 그러나 CUR 2.0 내보내기에서 리소스 태그 데이터를 계속 수신할 수 있습니다. 태그로 AWS 비용 및 사용량을 시각화하는 데

지원되는 QuickSight 대시보드를 원하는 경우 [AWS Well-Architected Labs의 CUDOS 대시보드](#)를 참조하세요. 현재는 레거시 CUR의 데이터만 사용하지만 향후 CUR 2.0을 지원할 예정입니다.

비용 및 사용 보고서 문제 해결

주제

- [비용 및 사용 보고서 데이터가 다른 Billing and Cost Management 특성에서 표시된 데이터와 일치하지 않는 이유는 무엇인가요?](#)
- [보고서 설정을 변경한 후 데이터를 어떻게 백필할 수 있나요?](#)
- [Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있는 이유는 무엇인가요?](#)
- [보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없는 이유는 무엇인가요?](#)
- [Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않는 이유는 무엇인가요?](#)
- [보고서에 포함된 열이 전월과 달라진 이유](#)
- [보고서의 열이 변경된 후 쿼리 또는 테이블이 작동하지 않는 이유는 무엇인가요?](#)
- [보고서를 쿼리하려면 어떻게 해야 하나요?](#)
- [Amazon EC2 전용 호스트의 결제 데이터는 어디에서 찾을 수 있나요?](#)
- [Amazon EC2 탄력적 IP 주소의 결제 데이터는 어떻게 해석해야 하나요?](#)
- [통합 결제에서 일반 요금과 혼합된 요금 또는 비용은 어떻게 다를까요?](#)
- [보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0 이유는 무엇인가요?](#)
- [보고서에서 전체 선결제 예약 인스턴스는 어떻게 분할 상환되나요?](#)

비용 및 사용 보고서 데이터가 다른 Billing and Cost Management 특성에서 표시된 데이터와 일치하지 않는 이유는 무엇인가요?

기타 결제 및 비용 관리 기능(Cost Explorer, 세부 결제 보고서, 결제 및 비용 관리 콘솔)에서는 다음과 같은 이유로 비용이 다르게 표시될 수 있습니다.

- 결제 기능은 다양한 방식으로 비용 데이터를 반올림합니다.
- 결제 기능마다 데이터 새로 고침 설정이 다를 수 있습니다. 예를 들어 비용 및 사용 보고서에서 청구가 확정된 후 적용된 환불, 크레딧 또는 지원 요금으로 이전 마감된 청구를 자동으로 새로 고칠지 여부를 선택할 수 있습니다. Cost Explorer는 동일한 항목을 자동으로 반영합니다. 이 시나리오에서 비용 및 사용 보고서에서 자동 새로 고침을 활성화하지 않으면 비용 및 사용 보고서 데이터가 Cost Explorer 데이터와 일치하지 않습니다.

- 결제 기능은 요금을 다양하게 그룹화할 수 있습니다. 예를 들어 결제 및 비용 관리 콘솔의 청구서 페이지에는 데이터 전송 요금이 AWS 서비스 요금 내에서 별도의 데이터 전송 그룹으로 표시됩니다. 한편, 비용 및 사용 보고서와 Cost Explorer에는 데이터 전송 요금이 각 서비스의 사용 유형으로 표시됩니다.

이러한 이유를 검토한 후에도 비용 및 사용 보고서와 다른 결제 및 비용 관리 기능 간에 여전히 불일치가 있다고 생각되면 지원 사례를 열어 비용 데이터 검토를 요청합니다. 지원 사례에는 보고서 이름과 검토할 결제 기간을 입력해야 합니다. 사례 열기에 대한 자세한 내용은 [내보내기 및 보고서 관련 도움말](#) 섹션을 참조하세요.

보고서 설정을 변경한 후 데이터를 어떻게 백필할 수 있나요?

지원 사례를 열어 비용 데이터 채우기를 요청합니다. 지원 사례에는 보고서 이름과 채울 결제 기간을 입력해야 합니다. 사례 열기에 대한 자세한 내용은 [내보내기 및 보고서 관련 도움말](#) 섹션을 참조하세요.

단, 다음과 같은 시나리오에서는 비용 데이터를 다시 채울 수 없습니다.

- 계정을 만든 날짜 이전의 비용 데이터는 다시 채울 수 없습니다.
- AWS Organizations 를 사용하고 관리 계정으로 지정된 계정과 같이 조직의 구조가 변경된 경우 이전 조직 구조로 데이터를 채울 수 없습니다.
- AWS Organizations 를 사용하고 조직을 변경하는 경우 현재 조직에 가입하기 전에에서 데이터 채우기를 가져올 수 없습니다.

Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있는 이유는 무엇인가요?

보고서의 보고서 경로 접두사에 / 문자가 있으면 Amazon S3 버킷에 이름 없는 폴더가 생성됩니다. 다음 보고서 업데이트에서 이름이 지정되지 않은 폴더를 제거하려면 보고서 설정을 편집하고 보고서 경로 접두사에서 / 문자를 제거합니다. 지침은 [비용 및 사용 보고서 구성 편집](#) 섹션을 참조하세요.

보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없는 이유는 무엇인가요?

보고서를 생성할 때 리소스 ID 포함 옵션을 선택할 수 있습니다. 보고서 버전을 기존 보고서 덮어쓰기로 설정한 상태에서 보고서를 생성하는 경우 보고서를 생성한 후에는 리소스 ID 포함 선택을 수정할 수 없습니다. 리소스 ID를 포함하려면 새 보고서를 만들고 리소스 ID 포함 옵션을 선택해야 합니다.

Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않는 이유는 무엇인가요?

Amazon Athena와 Amazon Redshift 데이터베이스는 비용 및 사용 보고서 열의 형식을 다르게 지정합니다. Amazon Athena는 열 이름(line_item_normalized_usage_amount)의 단어 사이에 밑줄을 추가합니다. Amazon Redshift는 열 유형과 속성(linitem_normalizeduseamount) 사이에 밑줄을 추가합니다. Amazon Athena 또는 Amazon Redshift의 열 이름 형식과 일치하도록 쿼리를 수정해야 합니다.

보고서에 포함된 열이 전월과 달라진 이유

가 보고서에 AWS 포함하는 열은 AWS 사용량에 따라 다릅니다. 모든 보고서에는 identity/, bill/, lineltem/ 접두사가 있는 열이 포함됩니다.

- identity/LineltemId
- identity/TimeInterval
- bill/InvoiceId
- bill/BillingEntity
- bill/BillType
- bill/PayerAccountId
- bill/BillingPeriodStartDate
- bill/BillingPeriodEndDate
- lineltem/UsageAccountId
- lineltem/LineltemType
- lineltem/UsageStartDate
- lineltem/UsageEndDate
- lineltem/ProductCode
- lineltem/UsageType
- lineltem/Operation
- lineltem/AvailabilityZone
- lineltem/ResourceId
- lineltem/UsageAmount
- lineltem/NormalizationFactor

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

다른 모든 열은 월별 AWS 사용량이 해당 열을 채우기 위한 데이터를 생성하는 경우에만 포함됩니다.

예를 들어, 보고서에는 해당 월에 절감형 플랜을 사용한 경우에만 savingsPlan/ 열이 포함됩니다.

보고서의 열이 변경된 후 쿼리 또는 테이블이 작동하지 않는 이유는 무엇인가요?

보고서에서 AWS 포함된 열은 해당 월의 AWS 사용량에 따라 달라집니다. 보고서에 포함된 열은 변경될 수 있으므로 보고서를 기반으로 하는 사용자 지정 쿼리나 표에서는 열 번호 대신 열 이름을 참조하는 것이 좋습니다.

보고서를 쿼리하려면 어떻게 해야 하나요?

비용 및 사용 보고서 쿼리에 대한 자세한 내용은 AWS Well-Architected Labs 웹 사이트의 [CUR 쿼리 라이브러리 도움말](#)을 참조하세요.

Amazon EC2 전용 호스트의 결제 데이터는 어디에서 찾을 수 있나요?

ResourceID 열에서 인스턴스 ID가 아닌 전용 호스트 ID를 찾습니다. 전용 호스트는 전용 호스트 실행 시간을 기준으로 측정되므로 보고서에는 호스트 ID와 관련된 측정 시간별로 전용 호스트 사용량이 표시됩니다.

Amazon EC2 탄력적 IP 주소의 결제 데이터는 어떻게 해석해야 하나요?

Amazon EC2 탄력적 IP 주소는 집계되어 측정됩니다. 즉, 보고서의 각 항목은 개별 탄력적 IP 주소와 일치하지 않습니다. 각 항목은 청구 가능한 총 시간을 나타냅니다. 실행 중인 인스턴스와 연결되어 있

는 탄력적 IP 주소는 한 개까지 무료로 사용할 수 있습니다. 인스턴스에 할당된 추가 탄력적 IP 주소 하나당 시간당 요금이 비례 할당됩니다. 또한 할당되지 않은 탄력적 IP 주소에 대해 시간당 요금이 AWS 부과됩니다.

통합 결제에서 일반 요금과 혼합된 요금 또는 비용은 어떻게 다를까요?

에 대한 통합 결제 AWS Organizations를 사용하면 혼합되지 않은 혼합 요금 또는 비용을 통해 독립 실행형 계정과 조직의 연결된 계정에 대한 계정 사용 비용을 이해하는 데 도움이 될 수 있습니다. 일부 서비스는 사용량 증가에 따라 단가를 낮출 수 있는 비용 계층을 제공합니다. 는 조직의 서비스에 대한 모든 사용량을 AWS 집계하므로 개별 계정은 조직의 월별 사용량에 사용량이 집계될 때 더 빨리 더 저렴한 티어에 액세스할 수 있습니다.

일반 요금은 개별 계정의 서비스 사용과 관련된 요금입니다. 항목의 경우 일반 비용은 사용량에 일반 요금을 곱한 값입니다. 독립 실행형 계정의 경우 일반 비용은 계정 사용 비용입니다. 혼합 요금은 조직의 총 사용량과 관련된 요금으로, 계정 전체의 평균을 구합니다. 항목의 경우 혼합 비용은 사용량에 혼합 요금을 곱한 값입니다. 혼합 비용은 조직의 연결 계정으로 계정을 사용하는 데 따른 비용입니다.

일반 및 혼합 비용 계산에 대한 자세한 정보 및 예제는 AWS Billing 사용 설명서의 [통합 청구서 이해](#)를 참조하세요.

보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0 이유는 무엇인가요?

예약형 인스턴스 할인이 적용되는 Amazon EC2 항목의 혼합 요금은 0입니다. 이러한 항목의 경우 LineltemType은 사용량 할인입니다.

혼합 비용은 사용량에 혼합 요금을 곱한 값입니다. 혼합 요금 또는 사용량 값이 0이면 혼합 비용도 0입니다.

보고서에서 전체 선결제 예약 인스턴스는 어떻게 분할 상환되나요?

모든 선결제 예약형 인스턴스는 전액 선결제로 청구되므로 분할 상환된 비용은 선결제 금액을 관련 기간(1년 또는 3년)으로 나누어 보고서에 반영됩니다.

reservation/AmortizedUpfrontCostForUsage 및 reservation/EffectiveCost는 모든 선결제 예약형 인스턴스의 요금과 동일합니다. 이는 두 항목 모두 예약형 인스턴스에 대한 선결제 금액을 계약 기간의 총 시간에 걸쳐 균등하게 분배하기 때문입니다.

RIFee가 0.00 USD이긴 하지만 보고서에는 모든 선결제 예약형 인스턴스에 대한 RIFee 항목이 입력되어 있을 것으로 예상됩니다. 이러한 항목은 해당 월의 시간당 기본 요금을 나타내며 다른 열에는 추가 사용 데이터가 있습니다. 모든 예약형 인스턴스는 RIFee 항목을 생성합니다.

탄소 배출량 데이터 내보내기 문제 해결

주제

- [IAM 권한으로 데이터 내보내기 및 CUR 2.0 테이블 사용이 가능함에도 불구하고, 탄소 배출량 테이블의 내보내기를 생성할 수 없는 이유는 무엇인가요?](#)
- [조직 내 일부 멤버 계정의 탄소 배출량 데이터를 볼 수 없는 이유는 무엇인가요?](#)
- [S3 버킷의 파일 중 하나가 비어 있는 이유는 무엇인가요?](#)
- [사용 데이터가 있는 경우 일부 리전 및 서비스에서 S3 내보내기에 탄소 배출량이 0으로 표시되는 이유는 무엇인가요?](#)
- [데이터 내보내기에서 과거 데이터 채우기를 탄소 배출량에 사용할 수 있나요?](#)
- [보고서 설정을 변경한 후 또는 새 방법론이 릴리스될 때 데이터를 채우려면 어떻게 해야 하나요?](#)
- [S3 버킷에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?](#)
- [내보내기에서 새로 추가된 열이 보이지 않는 이유는 무엇인가요?](#)
- [이전 방법론 버전을 사용하여 내 데이터가 제공되지 않는 이유는 무엇인가요?](#)

IAM 권한으로 데이터 내보내기 및 CUR 2.0 테이블 사용이 가능함에도 불구하고, 탄소 배출량 테이블의 내보내기를 생성할 수 없는 이유는 무엇인가요?

탄소 발자국 데이터에 액세스하려면 IAM 권한이 필요합니다.

다sustainability:GetCarbonFootprintSummary.

조직 내 일부 멤버 계정의 탄소 배출량 데이터를 볼 수 없는 이유는 무엇인가요?

관리(지급인) 계정을 사용하는 경우, 탄소 배출량 테이블에서 관리 계정 및 모든 멤버(사용자) 계정의 탄소 배출량 데이터를 자동으로 확인할 수 있습니다. 추가 구성이 필요하지 않습니다.

그러나 탄소 배출량 데이터의 데이터 지연은 최대 21일입니다. 새 멤버 계정의 경우 멤버 계정이 조직에 가입한 시기가 포함된 내보내기 기간까지 관리 계정의 탄소 배출량 데이터 내보내기에 데이터가 표시되지 않습니다. 예를 들어 1월에 새 멤버 계정을 연결한 경우 해당 데이터는 2월 내보내기에 먼저 표시됩니다.

마찬가지로 멤버 계정이 조직을 떠날 때 해당 데이터는 제거된 내보내기 기간까지 계속 표시됩니다.

S3 버킷의 파일 중 하나가 비어 있는 이유는 무엇인가요?

계정에 해당 월의 탄소 배출량 데이터가 없는 경우 해당 탄소 모델 버전 및 사용 기간에 대한 파일을 S3 버킷에 수신하지만 파일은 비어 있습니다.

사용 데이터가 있는 경우 일부 리전 및 서비스에서 S3 내보내기에 탄소 배출량이 0으로 표시되는 이유는 무엇인가요?

총 탄소 배출량이 0으로 표시되면 표시 임계값인 0.0000005 MTCO₂e보다 낮다는 의미입니다.

데이터 내보내기에서 과거 데이터 채우기를 탄소 배출량에 사용할 수 있나요?

예, 내보내기를 생성하면 2022년 1월까지 데이터를 다시 받게 되며, 그 이후에는 매월 첫 번째 전송과 한 달의 데이터가 전송됩니다. 계정이 2022년 1월 이후에 생성된 경우 계정 생성 날짜 이후부터 탄소 배출량 추정치를 받게 됩니다.

기존 데이터 내보내기가 있는 경우 채우기를 요청할 수 있습니다. 아래 질문에서 방법을 확인하세요.

보고서 설정을 변경한 후 또는 새 방법론이 릴리스될 때 데이터를 채우려면 어떻게 해야 하나요?

지원 사례를 열어 탄소 데이터 채우기를 요청합니다. 지원 사례에서는 보고서 이름과 채우기 시작 날짜를 제공해야 합니다. 사례 열기에 대한 자세한 내용은 [내보내기 및 보고서 도움말 받기를 참조하세요](#).

다음 시나리오에서는 탄소 데이터를 채울 수 없습니다.

- 계정을 생성한 날짜 이전에는 탄소 데이터에 대한 채우기를 가져올 수 없습니다.
- AWS Organizations를 사용하고 관리 계정으로 지정된 계정과 같이 조직의 구조가 변경된 경우 이전 조직 구조로 데이터를 채울 수 없습니다.
- AWS Organizations를 사용하고 조직을 변경하는 경우 현재 조직에 가입하기 전에에서 데이터 채우기를 가져올 수 없습니다.

S3 버킷에서 기록 데이터를 볼 수 없는 이유는 무엇인가요?

다음과 같은 이유로 S3 버킷에 기록 데이터가 누락되었을 수 있습니다.

- 과거 데이터 없음: 새 계정이 되거나 최근에 AWS Organizations의 멤버십이 변경되어 과거 탄소 배출량 추정치가 없는 계정이 있는 경우 S3 버킷에 과거 데이터를 채울 수 없습니다. 2022년 1월 이후에 계정이 생성된 경우, 계정이 활성화된 전체 기간에 대한 탄소 배출량 추정치를 받게 됩니다.
- 과거 데이터 채우기가 아직 진행 중임: Data Exports를 통한 과거 데이터 채우기를 완료하는 데 최대 24시간이 걸릴 수 있습니다. SDK/CLI를 사용하여 이 내보내기에 대한 ListExecutions API에서 실패한 채우기 실행이 있는지 또는 아직 진행 중인지 확인할 수 있습니다. 조금 더 기다리거나 ListExecutions를 사용하여 채우기가 진행 중이 아닌지 확인하세요.
- 과거 데이터 채우기 실패: 내부 오류로 인해 과거 데이터 채우기가 완료되지 않았을 수 있습니다. 24시간이 지나도 채우기가 완료되지 않은 경우 실패했다는 결론을 내릴 수 있습니다. 또는 SDK/CLI의 ListExecutions API를 사용하여 이 내보내기에 실패한 실행이 있는지 찾아볼 수 있습니다. 백필이 실패했다고 생각되면 새 내보내기를 생성해 봅니다. 두 번째로 실패하는 경우 AWS Support에 문의하는 것이 좋습니다.

내보내기에서 새로 추가된 열이 보이지 않는 이유는 무엇인가요?

기존 내보내기는 업데이트될 때까지 오리진 구성과 월별 업데이트를 유지합니다. 기존 내보내기에 새 열을 추가하려면 향후 월별 내보내기에 대한 내보내기 구성을 업데이트해야 합니다(이전에 내보낸 데이터는 변경되지 않음). 데이터를 새 열로 채우려면 새 내보내기를 생성해야 합니다. 최대 38개월의 기록 데이터와 월별 업데이트를 제공합니다.

이전 방법론 버전을 사용하여 내 데이터가 제공되지 않는 이유는 무엇인가요?

예상 배출량이 최대한 정확하도록 최신 방법론 버전을 사용하여 데이터를 게시합니다. 이전 버전의 데이터가 포함된 기존 데이터 내보내기가 있는 경우 삭제하지 않는 한 데이터를 계속 참조할 수 있습니다. 이전 버전을 검색하는 다른 방법은 없습니다.

데이터 내보내기의 파일 버전 관리를 덮어쓰기로 구성한 경우 동일한 방법론 버전 내에서 데이터 개선이 전달되는 경우에만 파일을 업데이트합니다. 데이터의 모든 기록 버전을 유지하려면 파일 버전 관리에서 새로 생성을 선택합니다. 이렇게 하면 데이터를 덮어쓰지 않습니다.

레거시 비용 및 사용량 보고서

이 섹션에서는 레거시 비용 및 사용량 보고서 기능에 대한 정보를 제공합니다.

Data Exports는 상세한 비용 및 사용량 데이터를 받을 수 있는 새롭고 권장되는 방법입니다 AWS. 자세한 내용은 [데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션](#)을 참조하세요.

AWS 비용 및 사용 보고서란 무엇입니까?

AWS 비용 및 사용 보고서(AWS CUR)에는 사용 가능한 가장 포괄적인 비용 및 사용 데이터 세트가 포함되어 있습니다. 비용 및 사용 보고서를 사용하여 소유한 Amazon Simple Storage Service(Amazon S3) 버킷에 AWS 결제 보고서를 게시할 수 있습니다. 시간, 일 또는 월별, 제품 또는 제품 리소스별 또는 직접 정의한 태그별로 비용을 구분하는 보고서를 받을 수 있습니다.는 하루에 한 번 버킷의 보고서를 쉼표로 구분된 값(CSV) 형식으로 AWS 업데이트합니다. Microsoft Excel, Apache OpenOffice Calc와 같은 스프레드시트 소프트웨어를 사용하여 보고서를 보거나 애플리케이션에서 Amazon S3 API를 사용하여 액세스할 수 있습니다.

AWS Cost and Usage Reports는 AWS 사용량을 추적하고 계정과 관련된 예상 요금을 제공합니다. 각 보고서에는 AWS 계정에서 사용하는 AWS 제품, 사용 유형 및 작업의 각 고유 조합에 대한 항목이 포함되어 있습니다. AWS 비용 및 사용 보고서를 사용자 지정하여 정보를 시간, 일 또는 월별로 집계할 수 있습니다.

AWS 비용 및 사용 보고서는 다음을 수행할 수 있습니다.

- Amazon S3 버킷에 보고서 파일 전송
- 보고서를 하루에 최대 세 번 업데이트합니다.
- AWS CUR API 참조를 사용하여 보고서 생성, 검색 및 삭제

비용 및 사용 보고서 작동 방식

비용 및 사용 보고서를 생성한 후에는 지정한 Amazon S3 버킷으로 보고서를 AWS 전송합니다.는 요금이 완료될 때까지 하루에 한 번 이상 보고서를 AWS 업데이트합니다.

보고서 파일에는.csv 파일 또는.csv 파일 모음과 매니페스트 파일이 포함됩니다. Amazon Athena, Amazon Redshift 또는 Quick과의 통합을 위해 보고서 데이터를 구성하도록 선택할 수 있습니다.

보고서 타임라인

보고서를 생성한 후 Amazon S3 버킷에 첫 번째 보고서를 AWS 전송하는 데 최대 24시간이 걸릴 수 있습니다.

전송이 시작된 후 하루에 한 번 이상 보고서 파일을 AWS 업데이트합니다. 해당 월의 각 보고서 업데이트는 누적되므로 보고서의 각 버전에는 해당 월의 모든 결제 데이터가 포함됩니다. 한 달 동안 받는 보고서 업데이트는 추정치입니다. AWS 서비스를 계속 사용함에 따라 요금이 변경될 수 있습니다.

Note

AWS 서비스마다 사용량 기반 결제 정보가 서로 다른 시간에 제공되므로 특정 시간 또는 날짜에 대한 업데이트가 서로 다른 시간에 수신되는 것을 알 수 있습니다.

AWS 는 결제 기간이 끝날 때까지 이전 보고서를 기반으로 구축됩니다.는 월말에 인보이스를 발행한 후 보고서의 사용 요금을 AWS 확정합니다. 보고서 결제 기간이 종료된 후는 이전 보고서의 정보가 없는 다음 달에 대한 새 보고서를 AWS 생성합니다.

보고서가 완료되면 해당 월 사용량에 환급, 크레딧 또는 AWS 지원 요금을 AWS 적용하는 경우 보고서를 AWS 업데이트할 수 있습니다. 개발자, 비즈니스 및 엔터프라이즈 지원은 최종 사용 요금을 기준으로 계산되므로 전월의 비용 및 사용 보고서에 대한 해당 월의 6일 또는 7일에 반영됩니다.는 계약 또는 계약 조건에 따라 크레딧 또는 환불을 AWS 적용합니다 AWS.

보고서 파일

보고서는 Amazon S3 버킷에 저장되는 .csv 파일 또는 .csv 파일 모음입니다. 보고서에서 생성하는 파일 수는 선택한 보고서 버전 관리 및 보고서 크기에 따라 달라집니다.

보고서를 생성하는 경우 새 보고서 버전을 만들거나 업데이트할 때마다 기존 보고서 버전을 덮어쓰도록 선택할 수 있습니다. 새 보고서 버전을 생성하도록 선택하면 보고서가 업데이트될 때마다 더 많은 파일이 생성됩니다.

개별 보고서의 크기가 1GB 이상으로 커지면 데스크톱 스프레드시트 애플리케이션의 용량을 초과하여 일부 행이 표시되지 않을 수 있습니다. 보고서가 대부분의 애플리케이션에서 처리할 수 있는 것보다 큰 경우(약 1백만 개의 행)는 보고서를 Amazon S3 버킷의 동일한 폴더에 저장된 여러 파일로 AWS 분할합니다.

AWS 또한는 별도의 파일로 환급을 생성합니다.는 월별 청구서 해지 후 환급을 AWS 발행합니다.

보고서 파일, 파일 이름 지정 규칙 및 버전 관리에 대한 자세한 내용은 [보고서 버전 이해](#) 섹션을 참조하세요.

보고서 열

각 보고서에는 AWS 비용 및 사용량에 대한 세부 정보가 포함된 여러 열이 포함되어 있습니다. 보고서에 AWS 포함된 열은 해당 월의 사용량에 따라 달라집니다.

모든 보고서에는 identity/, bill/, lineItem/ 접두사가 있는 열이 포함됩니다. 다른 모든 열은 월별 AWS 사용량이 해당 열을 채우기 위한 데이터를 생성하는 경우에만 포함됩니다.

예를 들어, 보고서에는 해당 월에 절감형 플랜을 사용한 경우에만 savingsPlan/ 열이 포함됩니다.

보고서의 열에 대한 자세한 정보는 [데이터 사전](#) 섹션을 참조하세요.

보고서 사용

Amazon S3 콘솔에서 보고서를 다운로드하거나, Amazon Athena를 사용하여 보고서를 쿼리하거나, Amazon Redshift 또는 Quick에 보고서를 업로드할 수 있습니다.

- Amazon S3 버킷을 생성하고 Athena를 사용하여 데이터를 쿼리하는 방법에 대한 자세한 정보는 [Amazon Athena를 사용하여 비용 및 사용 보고서 쿼리](#) 섹션을 참조하세요.
- Amazon Redshift에 업로드에 대한 자세한 정보는 [Amazon Redshift에 보고서 데이터 로드](#) 섹션을 참조하세요.
- Quick에 업로드하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Amazon Quick에 보고서 데이터 로드](#).

비용 및 사용 보고서 생성

Note

데이터 내보내기를 사용하면 비용 및 사용량 보고서(CUR) 2.0의 내보내기를 생성할 수 있습니다. 이는 상세한 비용 및 사용량 데이터를 받을 수 있는 새롭고 권장되는 방법입니다 AWS. 자세한 내용은 [데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션](#)을 참조하세요.

결제 전송을 사용하는 경우 레거시 페이지가 아닌 데이터 내보내기 AWS Cost and Usage Report 페이지에서만 결제 전송 보기에 대한 AWS Cost and Usage Report 보고서를 생성할 수 있습니다.

Billing and Cost Management 콘솔에서 비용 및 사용 보고서 페이지를 사용하여 비용 및 사용 보고서를 생성할 수 있습니다. 다음 섹션에서는 비용 및 사용 보고서를 시작하는 방법에 대한 정보를 제공합니다.

주제

- [비용 및 사용 보고서에 대한 Amazon S3 버킷 설정](#)
- [보고서 생성](#)

비용 및 사용 보고서에 대한 Amazon S3 버킷 설정

결제 보고서를 받으려면 보고서를 수신하고 저장할 Amazon S3 버킷이 AWS 계정에 있어야 합니다. 결제 콘솔에서 비용 및 사용 보고서를 생성할 때 소유하고 있는 기존 Amazon S3 버킷을 선택하거나 새 버킷을 생성할 수 있습니다. 어느 경우든 다음 기본 버킷 정책의 적용을 검토 및 확인하라는 메시지가 표시됩니다. Amazon S3 콘솔에서이 정책을 편집하거나 비용 및 사용 보고서를 생성한 후 버킷 소유자를 변경하면 AWS가 보고서를 전송할 수 없습니다. Amazon S3 버킷에 결제 보고서 데이터를 저장하면 표준 Amazon S3 요금이 결제됩니다. 자세한 내용은 [할당량 및 제한](#) 단원을 참조하십시오.

비용 및 사용 보고서를 생성할 때 모든 버킷에 다음 정책이 적용됩니다.

```
{
  "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}"
      }
    }
  }
]
}

```

이 기본 정책은 버킷 소유자가 비용 및 사용 보고서 데이터를 읽을 수 있도록 하고 비용 및 사용 보고서를 만든 계정이 해당 버킷을 소유하고 있음을 확인하는 데 도움이 됩니다. 구체적으로 설명하면 다음과 같습니다.

- 비용 및 사용 보고서가 전달될 때마다 AWS 먼저 보고서를 설정하는 계정이 버킷을 여전히 소유하고 있는지 확인합니다. 버킷 소유권이 변경된 경우 보고서는 전달되지 않습니다. 이를 통해 계정 결제 데이터의 보안을 보장할 수 있습니다. 이 버킷 정책은 AWS ("Effect": "Allow")가 버킷()을 소유한 계정을 확인할 수 있도록 허용합니다 "Action": ["s3:GetBucketAcl", "s3:GetBucketPolicy".
- Amazon S3 버킷에 보고서를 전송하려면 해당 버킷에 대한 쓰기 권한이 AWS 필요합니다. 이를 위해 버킷 정책은 ("Effect": "Allow")에게 소유한 버킷()에 ("Service": "billingreports.amazonaws.com") 보고서를 전달할 수 있는 AWS Cost and Usage Reports 서비스("Action": "s3:PutObject") 권한을 부여합니다 "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*".

이 버킷 정책은 전송된 후 비용 및 사용 보고서를 포함하여 버킷의 객체를 읽거나 삭제할 수 있는 AWS 권한을 부여하지 않습니다.

- ACL이 활성화된 Amazon S3 버킷의 경우는 보고서를 전송할 때 보고서에 BucketOwnerFullControl ACL을 AWS 추가로 적용합니다. 기본적으로 이러한 보고서와 같은 Amazon S3 객체는 해당 객체를 작성한 사용자 또는 서비스 주체만 읽을 수 있습니다. 사용자 또는 버킷 소유자에게 보고서를 읽을 수 있는 권한을 제공하려면 AWS가 BucketOwnerFullControl ACL을 적용해야 합니다. ACL은 버킷 소유자에게 이 보고서에 Permission.FullControl 권한을 부여합니다. 하지만 ACL을 비활성화하고 Amazon S3 버킷 정책을 사용하여 액세스를 제어하는 것

이 좋습니다. Amazon S3는 기본 설정을 변경했으며 새로 만든 버킷의 경우 기본적으로 ACL이 비활성화됩니다. 자세한 내용은 [객체 소유권 제어 및 버킷에 대해 ACL 사용 중지](#)를 참조하십시오.

비용 및 사용 보고서의 결제 콘솔에 잘못된 버킷 오류가 표시되면 보고서 설정 후 이 정책과 버킷 소유권이 변경되지 않았는지 확인해야 합니다.

보고서 생성

Note

데이터 내보내기를 사용하면 비용 및 사용량 보고서(CUR) 2.0의 내보내기를 생성할 수 있습니다. 이는 상세한 비용 및 사용량 데이터를 받을 수 있는 새롭고 권장되는 방법입니다. 자세한 내용은 [데이터 내보내기 CUR에서 CUR 2.0으로 마이그레이션](#)을 참조하세요.

Billing and Cost Management 콘솔의 비용 및 사용 보고서 페이지를 사용하여 비용 및 사용 보고서를 생성할 수 있습니다. 개별 AWS 계정에 대해 최대 10개의 보고서를 생성할 수 있습니다.

Note

가 Amazon S3 버킷 AWS 에 보고서를 전송하기 시작하는 데 최대 24시간이 걸릴 수 있습니다. 전송이 시작된 후에는 AWS 비용 및 사용 보고서 파일을 하루에 한 번 이상 AWS 업데이트합니다.


비용 및 사용 보고서 생성

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 보고서 생성을 선택합니다.
4. 보고서 이름에 보고서 이름을 입력합니다.
5. 보고서 추가 콘텐츠에서 리소스 ID 포함을 선택하여 보고서에 각 개별 리소스의 ID를 포함합니다.

Note

리소스 ID를 포함하면 각 리소스에 대한 개별 항목이 생성됩니다. 이를 통해 AWS 사용량에 따라 비용 및 사용 보고서 파일 크기가 크게 늘어날 수 있습니다.

- 공유 리소스에 대한 자세한 비용 및 사용량을 포함하려면 비용 할당 데이터 분할을 선택합니다 (Amazon ECS 및 Amazon EKS).

 Note

비용 할당 데이터 분할을 포함하면 각 리소스(즉, ECS 작업 및 Kubernetes 포드)에 대한 개별 항목이 생성됩니다. 이렇게 하면 AWS 사용량에 따라 비용 및 사용 보고서 파일의 크기가 크게 증가할 수 있습니다.

- 데이터 새로 고침 설정에서가 청구서를 완료한 후 계정에 환불, 크레딧 또는 지원 요금을 AWS 적용하는 경우 AWS 비용 및 사용 보고서를 새로 고칠지 여부를 선택합니다. 보고서가 새로 고침되면 새 보고서가 Amazon S3에 업로드됩니다.
- 다음을 선택합니다.
- S3 버킷에서 구성을 선택합니다.
- S3 버킷 구성 대화 상자에서 다음 중 하나를 수행합니다.
 - 기존 버킷을 선택합니다.
 - 버킷 생성을 선택하고 버킷 이름을 입력한 후 새 버킷을 생성할 리전을 선택합니다.
- 버킷 정책을 검토하고 다음 기본 정책이 버킷에 적용됨을 선택한 다음 저장을 선택합니다.
- 보고서 경로 접두사에 보고서 이름의 접두어가 되는 보고서 경로 접두사를 입력합니다.
- 시간 세부 수준에 대해 다음 중 하나를 선택합니다.
 - 시간별: 보고서의 항목을 시간별로 집계하려면 선택합니다.
 - 일별: 보고서의 항목을 일별로 집계하려면 선택합니다.
 - 월별 보고서의 항목을 월별로 집계하려는 경우.
- 보고서 버전 관리에서, 보고서의 각 버전을 이전 버전을 겹쳐쓸지 또는 이전 버전과 별도로 추가 제공할지를 선택합니다.

보고서를 덮어쓰면 Amazon S3 스토리지 비용을 절약할 수 있습니다. 새 보고서 버전을 제공하면 시간이 지남에 따라 결제 데이터의 감사 가능성을 개선할 수 있습니다.
- 보고서 데이터 통합에서 비용 및 사용 보고서를 Amazon Athena, Amazon Redshift 또는 Quick과 통합할지 여부를 선택합니다. 보고서는 다음 형식으로 압축됩니다.
 - Athena: parquet 형식
 - Amazon Redshift 또는 Quick: .gz 압축
- 다음을 선택합니다.

17. 보고서 설정을 검토했으면 [Review and Complete]를 선택합니다.

Billing and Cost Management 콘솔에서 비용 및 사용 보고서 페이지로 언제든지 돌아가서 보고서가 마지막으로 업데이트된 시점을 확인할 수 있습니다.

보고서 보기 및 관리

비용 및 사용 보고서의 정보를 보려면 결제 및 비용 관리 콘솔을 사용합니다. Amazon S3 콘솔을 사용하여 보고서 파일을 볼 수 있습니다.

다음 절차를 사용하여 보고서 및 보고서 파일을 찾을 수 있습니다.

보고서 세부 정보 및 파일 보기

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 보고서 목록에서 보려는 보고서 이름을 선택합니다.
4. 보고서 세부 정보 페이지에서 보고서 설정을 볼 수 있습니다.
5. 보고서 파일을 보려면 보고서 세부 정보 페이지의 보고서 경로 접두사를 기록해 둡니다.
6. Amazon S3 버킷에 나열된 버킷 이름을 선택합니다. 이 링크는 Amazon S3 콘솔에서 이 버킷을 엽니다.
7. 버킷의 객체 목록에서, 5단계에서 기록한 보고서 경로 접두사의 첫 부분이 있는 폴더를 선택합니다. 예를 들어 보고서 경로 접두사가 **example-report-prefix/example-report-name**인 경우 example-report-prefix라는 폴더를 선택합니다.
8. 폴더의 객체 목록에서, 5단계에서 기록한 보고서 경로 접두사의 두 번째 부분이 있는 폴더를 선택합니다. 예를 들어 보고서 경로 접두사가 **example-report-prefix/example-report-name**인 경우 example-report-name이라는 폴더를 선택합니다. 이 폴더에는 보고서 파일이 들어 있습니다.

최신 보고서 버전 보기

AWS 는 요금이 완료될 때까지 하루에 한 번 이상 비용 및 사용 보고서를 업데이트합니다. 보고서를 생성하는 경우 새 보고서 버전을 만들거나 업데이트할 때마다 기존 보고서 버전을 덮어쓰도록 선택할 수 있습니다.

업데이트할 때마다 새 보고서 버전을 생성하도록 보고서를 구성한 경우 매니페스트 파일의 `assemblyId`를 사용하여 최신 보고서 파일을 찾습니다.

보고서 버전이 여러 개 있을 때 최신 보고서 파일 보기

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 보고서 목록에서 보려는 보고서 이름을 선택합니다.
4. 보고서 세부 정보 페이지에서 보고서 경로 접두사를 기록해 둡니다.
5. Amazon S3 버킷에 나열된 버킷 이름을 선택합니다. 이 링크는 Amazon S3 콘솔에서 이 버킷을 엽니다.
6. 버킷의 객체 목록에서, 4단계에서 기록한 보고서 경로 접두사의 첫 부분이 있는 폴더를 선택합니다. 예를 들어 보고서 경로 접두사가 **example-report-prefix/example-report-name**인 경우 `example-report-prefix`라는 폴더를 선택합니다.
7. 폴더의 객체 목록에서 4단계에서 기록해 둔 보고서 경로 접두사의 두 번째 부분으로 이름이 지정된 폴더를 선택합니다. 예를 들어 보고서 경로 접두사가 **example-report-prefix/example-report-name**인 경우 `example-report-name`이라는 폴더를 선택합니다.
8. 가장 최근 결제 기간으로 이름이 지정된 폴더를 엽니다(YYYYMMDD-YYYYMMDD 형식).
9. **example-report-name-Manifest.json** 파일을 엽니다.
10. 매니페스트 파일 상단에 있는 `assemblyId`를 기록해 둡니다. `assemblyId` 값은 최신 보고서 파일이 있는 폴더의 이름에 해당합니다.
11. 최근 결제 기간으로 이름이 지정된 폴더가 표시된 Amazon S3 콘솔 페이지로 돌아갑니다.
12. 10단계에서 기록해 둔 `assemblyId` 값으로 이름이 지정된 폴더를 엽니다. 예를 들어, `assemblyId` 값이 **20210129T123456Z**이면 `20210129T123456Z/`라는 폴더를 엽니다. 이 폴더에는 최신 보고서 파일이 들어 있습니다.

최종 보고서 보기

월말에 인보이스를 발행하면가 보고서의 사용 요금을 AWS 종료합니다. 보고서의 항목이 최종 항목인지 확인하려면 `bill/InvoiceId` 열을 검토합니다. 행 항목이 최종 항목인 경우 `bill/InvoiceId` 열에 AWS 인보이스 ID가 채워집니다. 항목이 아직 최종 항목이 아닌 경우 `bill/InvoiceId` 열은 비어 있습니다.

전체 보고서가 확정되었는지 확인하려면 `bill/InvoiceId` 열을 검토합니다. 보고서가 확정인 경우 `bill/InvoiceId` 열에 청구서 ID 값이 채워집니다. 보고서가 아직 확정되지 않은 경우 `bill/InvoiceId` 열은 비어 있습니다.

Note

보고서가 완료되면 해당 월 사용량에 환급, 크레딧 또는 지원 요금을 AWS 적용하는 경우 보고서를 AWS 업데이트할 수 있습니다. 개발자, 비즈니스 및 엔터프라이즈 지원은 최종 사용 요금을 기준으로 계산되므로 전월 보고서의 경우 해당 월의 6일 또는 7일에 반영됩니다.는 계약 또는 계약 조건에 따라 크레딧 또는 환불을 AWS 적용합니다 AWS.

보고서 버전 이해

AWS 는 요금이 완료될 때까지 하루에 한 번 이상 비용 및 사용 보고서를 업데이트합니다. 보고서를 생성하는 경우 새 보고서 버전을 만들거나 업데이트할 때마다 기존 보고서 버전을 덮어쓰도록 선택할 수 있습니다.

보고서 파일에는 .csv 파일 또는 .csv 파일 모음과 매니페스트 파일이 포함됩니다. 보고서에는 Amazon Athena, Amazon Redshift 또는 Quick과 데이터 통합을 지원하는 추가 파일도 포함될 수 있습니다.

다음 섹션에서는 선택한 보고서 버전 관리에 따른 파일 구성 및 이름 지정 규칙을 설명합니다.

비용 및 사용 보고서 전송 타임라인

보고서 기간 동안는가 결제 기간이 끝날 때까지 이전 보고서에 대한 report. AWS builds를 AWS 업데이트할 때마다 새 보고서와 새 매니페스트 파일을 AWS 전송합니다. 보고서 결제 기간이 종료된 후는 이전 보고서의 정보가 없는 새 보고서를 AWS 생성합니다.

새로운 비용 및 사용 보고서 버전 생성

이전 비용 및 사용 보고서를 유지하도록 선택하면 AWS CUR은 다음과 같은 Amazon S3 조직 및 명명 규칙을 사용합니다.

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

- report-prefix = 보고서에 할당하는 접두사입니다.
- report-name = 보고서에 할당하는 이름입니다.
- yyyymmdd-yyyymmdd = 보고서에 해당하는 날짜 범위입니다. 보고서는 날짜 범위 말에 최종 완성됩니다.
- assemblyId = 보고서가 업데이트될 때마다 AWS 생성하는 ID입니다.

- file-number = 업데이트에 대용량 파일이 포함된 경우 여러 파일로 분할할 수 AWS 있습니다. file-number는 업데이트에서 다른 파일을 추적합니다.
- csv = 보고서 파일의 포맷입니다.
- zip 또는 gz = 압축 유형이 보고서 파일에 적용됩니다.

예를 들어 보고서는 다음 파일의 컬렉션으로 제공될 수 있습니다.

```
<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 는 보고서 날짜 범위의 모든 보고서를 동일한 report-prefix/report-name/yyyymmdd-yyyyymmdd 폴더에 전송합니다.는 각 보고서에 고유한 ID를 AWS 부여하고 날짜 범위 폴더의 assemblyId 하위 폴더에 전달합니다. 보고서가 한 개 파일로 생성할 수 없을 만큼 클 경우 여러 개의 파일로 분할된 다음 같은 assemblyId 폴더로 전송됩니다.

이전 보고서를 유지할 때 파일 매니페스트 파일 대한 자세한 내용은 [비용 및 사용 보고서 매니페스트 파일](#) 단원을 참조하십시오.

이전 비용 및 사용 보고서 덮어쓰기

이전 비용 및 사용 보고서를 덮어쓰도록 선택하면 AWS CUR은 다음 Amazon S3 조직 및 이름 지정 규칙을 사용합니다.

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

- report-prefix = 보고서에 할당하는 접두사입니다.
- report-name = 보고서에 할당하는 이름입니다.
- yyyyymmdd-yyyyymmdd = 보고서에 해당하는 날짜 범위입니다. AWS 에서 날짜 범위 말에 보고서를 최종 완성합니다.

-
- file-number = 업데이트에 대용량 파일이 포함된 경우 여러 파일로 분할할 수 AWS 있습니다. file-number는 업데이트에서 다른 파일을 추적합니다.
- csv = 보고서 파일의 포맷입니다.
- zip 또는 gz = 압축 유형이 보고서 파일에 적용됩니다.

예를 들어 보고서는 다음 파일의 컬렉션으로 제공될 수 있습니다.

```
<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 사양

AWS CUR을 생성할 때 Athena 지원을 선택한 경우 파일 이름 지정 규칙은 형식 및 압축을 제외하고 AWS CUR을 덮어쓰도록 선택할 때와 동일합니다. Athena AWS CUR 파일은 .parquet 대신을 사용합니다. 예를 들어 보고서는 다음 파일의 컬렉션으로 제공될 수 있습니다.

```
<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
```

CloudFormation 사양

CUR 파일 외에도 AWS 는 Athena를 사용하여 Amazon S3 데이터를 쿼리할 수 있는 CloudFormation 스택을 설정하는 데 사용할 수 있는 CloudFormation 템플릿을 AWS 제공합니다. CloudFormation 템플릿을 사용하지 않으려면 제공된 SQL을 사용하여 자체 Athena 테이블을 생성할 수 있습니다. 자세한 내용은 [Amazon Athena를 사용하여 비용 및 사용 보고서 쿼리](#) 단원을 참조하십시오.

비용 및 사용 보고서 매니페스트 파일

가 AWS CUR을 AWS 업데이트하면 Amazon Athena, Amazon Redshift 또는 Quick에 사용할 수 있는 매니페스트 파일 AWS 도 생성하고 전송합니다.

매니페스트 파일은 이름 지정 규칙을 사용하며 다음을 나열합니다.

- 현재까지 보고서에 포함된 모든 세부 정보 열
- 보고서가 여러 파일로 분할된 경우 보고서 파일 목록
- 보고서에 적용되는 기간 및 기타 정보

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

새로운 비용 및 사용 보고서 버전 생성

이전 비용 및 사용 보고서를 유지할 때 매니페스트 파일은 날짜 범위 폴더 및 assemblyId 폴더 모두에 전달됩니다. 는 날짜 범위에 대한 새 AWS CUR을 AWS 생성할 때마다 날짜 범위 폴더에 저장된 매니페스트 파일을 업데이트된 매니페스트 파일로 덮어씁니다. 는 동일한 업데이트된 매니페스트 파일을 해당 업데이트의 파일과 함께 assemblyId 폴더에 AWS 전달합니다. assemblyId 폴더의 매니페스트 파일에는 덮어쓰지 않습니다.

이전 비용 및 사용 보고서 덮어쓰기

이전 AWS CUR을 덮어쓰면 매니페스트 파일이 month=mm 폴더로 전달됩니다. 매니페스트 파일은 보고서 파일과 함께 덮어씁니다.

Amazon Redshift 사양

AWS CUR에서 Amazon Redshift 지원 옵션을 선택한 경우는 보고서를 Amazon Redshift에 업로드하는 데 필요한 SQL 명령이 포함된 파일을 AWS 생성하고 전송합니다. SQL 파일은 일반 텍스트 편집기를 사용하여 열 수 있습니다. SQL 파일은 다음 명명 규칙을 사용합니다.

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

RedshiftCommands 파일의 명령을 사용할 경우 RedshiftManifest 파일을 열지 않아도 됩니다.

Important

manifest 파일에 따라 copy 파일의 RedshiftCommands 명령이 업로드하는 보고서 파일이 결정됩니다. manifest 파일을 삭제하거나 제거할 경우 RedshiftCommands 파일의 copy 명령이 중단됩니다.

Amazon Athena 사양

AWS CUR에서 Amazon Athena 지원 옵션을 선택한 경우 AWS 는 필요한 모든 리소스를 설정하는 데 도움이 되는 여러 파일을 생성하고 전송합니다.는 CloudFormation 템플릿, Athena 테이블을 수동으로 생성하는 SQL 파일, AWS CUR 새로 고침 상태를 확인하는 SQL이 포함된 파일을 AWS 제공합니다. 이러한 파일은 다음 명명 규칙을 사용합니다.

```
<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>
```

비용 및 사용 보고서 구성 편집

Billing and Cost Management 콘솔의 비용 및 사용 보고서 페이지를 사용하여 비용 및 사용 보고서를 편집할 수 있습니다.

Note

보고서 이름은 편집할 수 없습니다. 보고서 버전 관리를 위해 덮어쓰기를 선택한 경우 보고서에 리소스 ID, 시간 세분화 또는 보고서 버전 관리가 포함되는지 여부와 상관없이 보고서 이름을 편집할 수 없습니다. 덮어쓰기로 설정된 보고서를 삭제하고 이름, Amazon S3 버킷 및 경로 접두사가 동일한 새 보고서를 생성하는 경우 데이터가 손상되고 정확하지 않을 수 있습니다.

비용 및 사용 보고서 편집

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.

2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 삭제하려는 보고서를 선택하고 Edit report(보고서 편집)을 선택합니다.
4. (버전이 지정된 보고서에만 해당) 보고서 추가 콘텐츠에서 리소스 ID 포함을 선택하여 보고서에 각 개별 리소스의 ID를 포함합니다.

Note

리소스 ID를 포함하면 각 리소스에 대한 개별 항목이 생성됩니다. 이를 통해 AWS 사용량에 따라 비용 및 사용 보고서 파일 크기가 크게 늘어날 수 있습니다.

5. 공유 리소스에 대한 자세한 비용 및 사용량을 포함하려면 비용 할당 데이터 분할을 선택합니다 (Amazon ECS 및 Amazon EKS).

Note

비용 할당 데이터 분할을 포함하면 각 리소스(즉, ECS 작업 및 Kubernetes 포드)에 대한 개별 항목이 생성됩니다. 이렇게 하면 AWS 사용량에 따라 비용 및 사용 보고서 파일의 크기가 크게 증가할 수 있습니다.

6. 데이터 새로 고침 설정에서가 청구서를 완료한 후 계정에 환불, 크레딧 또는 지원 요금을 AWS 적용하는 경우 AWS 비용 및 사용 보고서를 새로 고칠지 여부를 선택합니다. 보고서가 새로 고침되면 새 보고서가 Amazon S3에 업로드됩니다.
7. 다음을 선택합니다.
8. S3 버킷의 경우 보고서를 전송할 Amazon S3 버킷의 이름을 입력합니다.
9. 확인을 선택합니다.

Note

버킷에는 유효하고 적절한 사용 권한이 있어야 합니다. 버킷에 권한 추가에 대한 자세한 정보는 [Amazon Simple Storage Service 사용 설명서의 버킷 및 객체 액세스 권한 설정](#)을 참조하세요.

10. 보고서 경로 접두사에 보고서 이름의 접두어가 되는 보고서 경로 접두사를 입력합니다.
11. (버전이 지정된 보고서에만 해당) 시간 세부 수준에 대해 다음 중 하나를 선택합니다.
 - 시간별: 보고서의 항목을 시간별로 집계하려면 선택합니다.
 - 일별: 보고서의 항목을 일별로 집계하려면 선택합니다.

- 월별 보고서의 항목을 월별로 집계하려는 경우.
12. (버전이 지정된 보고서에 한함) 보고서 버전 관리에서 보고서의 각 버전이 이전 버전을 덮어쓰거나 이전 버전에 추가로 전송할지 여부를 선택합니다.
 13. 보고서 데이터 통합에서 AWS CUR을 Amazon Athena, Amazon Redshift 또는 Quick과 통합할지 여부를 선택합니다. 보고서는 다음 형식으로 압축됩니다.
 - Athena: Parquet 형식
 - Amazon Redshift 또는 Quick: .gz 압축
 14. 저장을 선택합니다.

에 대한 비용 및 사용 보고서 사용 AWS Organizations

에서 관리 계정과 멤버 계정 AWS Organizations 모두 비용 및 사용 보고서를 생성할 수 있습니다. 보고서 생성 기능을 허용하거나 제한하는 IAM 정책은 두 계정 유형 모두에 동일합니다.

Note

비용 및 사용 보고서를 생성하는 계정은 AWS가 보고서를 보내는 Amazon S3 버킷도 소유해야 합니다. 다른 계정이 소유한 Amazon S3 버킷으로 비용 및 사용 보고서를 전송하도록 구성할 수 없습니다. S3 버킷의 설정 요구 사항에 대한 자세한 정보는 [비용 및 사용 보고서에 대한 Amazon S3 버킷 설정](#) 섹션을 참조하세요.

멤버 계정으로서의 비용 및 사용 보고서 관리

조직 내 멤버 계정에 대한 비용 및 사용 보고서를 만들 권한이 있는 경우 멤버 계정의 비용 및 사용 데이터에 대한 보고서만 만들 수 있습니다. 멤버 계정은 해당 계정이 현재 조직의 멤버로 활동한 기간 동안의 비용 및 사용량에 대한 보고서를 받습니다.

매월 15일에 조직 A를 떠나 조직 B에 가입하는 멤버 계정을 예로 들어 보겠습니다. 그러면 멤버 계정이 보고서를 생성합니다. 멤버 계정이 조직 B에 가입한 후 보고서를 만들었기 때문에 해당 월의 멤버 계정 보고서에는 해당 계정이 조직 B의 멤버로 활동한 기간에 대한 결제 데이터만 포함됩니다.

멤버 계정이 새 조직에 가입하면 멤버 계정의 비용 및 사용량이 새 조직의 보고서에 기록됩니다. 이는 멤버 계정으로 전환되어 새 조직에 가입하는 관리 계정의 경우와 동일한 결과입니다.

멤버 계정이 조직을 탈퇴하거나 독립 실행형 계정으로 전환되는 경우, 멤버 계정은 이전 보고서가 저장되는 Amazon S3 버킷에 대한 권한이 있는 한 이전 보고서에 계속 액세스할 수 있습니다.

관리 계정으로서의 비용 및 사용 보고서 관리

AWS Organizations 관리 계정의 관리자이고 멤버 계정이 보고서를 생성하지 않도록 하려면 멤버 계정이 보고서를 생성하지 못하도록 하는 서비스 제어 정책(SCP)을 적용할 수 있습니다. SCP는 멤버 계정이 새 보고서를 만드는 것을 방지할 수 있지만 이전에 만든 보고서는 삭제하지 않습니다.

Note

SCP는 멤버 계정에만 적용할 수 있습니다. 관리 계정이 보고서를 생성하지 못하게 하려면 관리 계정의 사용자 역할에 연결된 IAM 정책을 수정합니다.

통합 결제에 대한 자세한 정보는 AWS Billing 사용 설명서의 [AWS Organizations에 대한 통합 결제](#)를 참조하세요.

Amazon Athena를 사용하여 비용 및 사용 보고서 쿼리

Amazon Athena는 표준 SQL을 사용하여 Amazon Simple Storage Service(Amazon S3)의 AWS 비용 및 사용 보고서(AWS CUR)에서 데이터를 분석하는 데 사용할 수 있는 서버리스 쿼리 서비스입니다. 이렇게 하면 CUR 데이터를 쿼리하기 위해 자체 데이터 AWS 웨어하우스 솔루션을 생성할 필요가 없습니다.

Athena와 함께 사용할 새 Amazon S3 버킷과 새 AWS CUR 보고서를 모두 생성하는 것이 좋습니다. AWS CUR은 Athena에 대한 Apache Parquet 압축 형식만 지원하고 S3 버킷에 저장된 이전 보고서를 자동으로 덮어씁니다.

이 섹션에서는 Athena를 비용 및 사용 보고서와 함께 사용하는 방법을 설명합니다. Athena 서비스에 대한 전체 설명은 [Amazon Athena 사용 설명서](#)를 참조하세요.

주제

- [CloudFormation 템플릿을 사용하여 Athena 설정](#)
- [Athena 수동 설정](#)
- [Amazon Athena 쿼리 실행](#)
- [보고서 데이터를 다른 리소스로 로드](#)

Athena를 사용하여 보고서를 쿼리하는 방법에 대한 데모는 다음 동영상을 참조하세요.

Amazon Athena를 사용하여 비용 및 사용 보고서 분석

CloudFormation 템플릿을 사용하여 Athena 설정

⚠ Important

CloudFormation 는 리전 간 리소스를 지원하지 않습니다. CloudFormation 템플릿을 사용하려는 경우 동일한 AWS 리전에 모든 리소스를 생성해야 합니다. 리전은 다음 서비스를 지원해야 합니다.

- AWS Lambda
- Amazon Simple Storage Service(Amazon S3)
- AWS Glue
- Amazon Athena

비용 및 사용 보고서와 Athena의 통합을 간소화하고 자동화하기 위해서는 Athena 통합을 위해 설정한 보고서와 함께 여러 주요 리소스가 포함된 CloudFormation 템플릿을 AWS 제공합니다. CloudFormation 템플릿에는 AWS Glue 크롤러, AWS Glue 데이터베이스 및 AWS Lambda 이벤트가 포함됩니다.

를 사용하는 Athena 통합 설정 프로세스는 버킷에 이미 있을 수 있는 모든 Amazon S3 이벤트를 CloudFormation 제거합니다. 이는 기존 AWS CUR 보고서에 대해 보유한 기존 이벤트 기반 프로세스에 부정적인 영향을 미칠 수 있습니다. Athena와 함께 사용할 새 Amazon S3 버킷과 새 AWS CUR 보고서를 모두 생성하는 것이 좋습니다.

CloudFormation 템플릿을 사용하여 Athena 통합을 자동화하려면 다음을 수행해야 합니다.

- 보고서에 대한 새 Amazon S3 버킷을 생성합니다. 자세한 정보는 Amazon S3 사용 설명서의 [버킷 생성](#)을 참조하세요.
- Athena와 함께 사용할 [새 보고서를 생성](#)합니다. 설정 프로세스 중 보고서 데이터 통합을 위해 Athena를 선택합니다.
- Amazon S3 버킷에 첫 번째 보고서가 전송될 때까지 기다립니다. 가 첫 번째 보고서를 전송하는 데 최대 24시간 AWS 이 걸릴 수 있습니다.

Athena CloudFormation 템플릿을 사용하려면

1. <https://console.aws.amazon.com/s3/>에서 S3 콘솔을 엽니다.
2. 버킷 목록에서 AWS CUR 보고서를 수신하도록 선택한 버킷을 선택합니다.

3. 보고서 경로 접두사(*your-report-path-prefix/*)를 선택합니다. 그런 다음 보고서 이름 (*your-report-name/*)을 선택합니다.
4. .yaml 템플릿 파일을 선택합니다.
5. 객체 작업을 선택한 후 다음으로 다운로드를 선택합니다.
6. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
7. CloudFormation 이전예를 사용한 적이 없는 경우 새 스택 생성을 선택합니다. 그렇지 않은 경우 스택 생성을 선택합니다.
8. Prepare template(템플릿 준비)에서 Template is ready(템플릿 준비가 완료되었습니다)를 선택합니다.
9. Template source(템플릿 소스)에서 템플릿 파일 업로드를 선택합니다.
10. 파일 선택을 선택합니다.
11. 다운로드한 .yaml 템플릿을 선택한 다음 열기를 선택합니다.
12. 다음을 선택합니다.
13. 스택 이름에 템플릿 이름을 입력하고 다음을 선택합니다.
14. 다음을 선택합니다.
15. 페이지 하단에서 가 IAM 리소스를 생성할 AWS CloudFormation 수 있음을 승인합니다를 선택합니다.

이 템플릿은 다음 리소스를 만듭니다.

- 3가지 IAM 역할
- AWS Glue 데이터베이스
- AWS Glue 크롤러
- 2가지 Lambda 함수
- Amazon S3 알림

16. 스택 생성을 선택합니다.

기존 Athena CloudFormation 템플릿을 업데이트하려면

1. <https://console.aws.amazon.com/s3>에서 S3 콘솔을 엽니다.
2. 버킷 목록에서 AWS CUR 보고서를 수신하도록 선택한 버킷을 선택합니다.
3. 보고서 경로 접두사(*your-report-path-prefix/*)를 선택합니다. 그런 다음 보고서 이름 (*your-report-name/*)을 선택합니다.

4. .yml 템플릿 파일을 선택합니다.
5. 객체 작업을 선택한 후 다음으로 다운로드를 선택합니다.
6. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
7. 이전에 만든 스택을 선택한 다음 업데이트를 선택합니다.
8. Prepare template(템플릿 준비)에서 Replace current template(현재 템플릿 바꾸기)을 선택합니다.
9. Template source(템플릿 소스)에서 템플릿 파일 업로드를 선택합니다.
10. 파일 선택을 선택합니다.
11. 다운로드한 .yml 템플릿을 선택한 다음 열기를 선택합니다.
12. 다음을 선택합니다.
13. 스택 세부 정보 지정 페이지에서 세부 정보를 수정하고 다음을 선택합니다.
14. 다음을 선택합니다.
15. 페이지 하단에서 이 IAM 리소스를 생성할 AWS CloudFormation 수 있음을 승인합니다를 선택합니다.
16. 스택 업데이트를 선택합니다.

Athena 수동 설정

직접 테이블을 생성하는 대신 AWS CloudFormation 템플릿을 사용하여 테이블을 생성하는 것이 좋습니다. 제공된 SQL 쿼리는 한 달의 데이터만 포함하는 테이블을 생성하지만 AWS CloudFormation 템플릿은 여러 달을 포함할 수 있고 자동으로 업데이트되는 테이블을 생성합니다. AWS CloudFormation 템플릿을 설정하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [the section called “CloudFormation으로 Athena 설정”](#).

CloudFormation 템플릿을 사용하여 Athena 테이블을 설정하지 않도록 선택한 경우 아래 단계를 수동으로 따릅니다. AWS CUR 데이터에 SQL 쿼리를 실행하려면 먼저 테이블을 생성해야 합니다. 이 단계는 한 달에 한 번 이상 수행해야 하며 테이블에는 현재 AWS CUR의 데이터만 포함됩니다.

테이블 생성 프로세스의 일부로는 AWS CUR 열 이름을 AWS 변환합니다. 변환 프로세스에 대한 자세한 내용은 [the section called “열 이름”](#) 단원을 참조하십시오.

- [Athena 테이블 생성](#)
- [비용 및 사용 보고서](#)
- [보고서 파티션 업로드](#)

Athena 테이블 생성

AWS에는 AWS CUR 버킷에서 테이블을 생성하기 위해 실행해야 하는 SQL이 포함되어 있습니다.

Athena 테이블 생성

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. 버킷 목록에서 비용 및 사용 보고서를 수신하려고 선택한 버킷을 선택합니다.
3. *your-report-prefix-your-report-name-path-to-report* 경로를 탐색합니다.

정확한 경로는 AWS CUR이 이전 버전을 덮어쓰도록 설정되어 있는지 여부에 따라 달라집니다. 자세한 내용은 [비용 및 사용 보고서 전송 타임라인](#) 단원을 참조하십시오.

4. *my-report-name-create-table.sql* 파일을 엽니다.
5. CREATE로 시작하여 LOCATION '*s3://your-report-prefix/your-report-name/the-rest-of-the=path*'로 끝나는 SQL을 파일에서 복사합니다. Athena 데이터베이스를 생성하려면 데이터베이스 이름과 테이블이 필요하기 때문에 첫 번째 줄을 기록합니다.
6. <https://console.aws.amazon.com/athena/>에서 Athena 콘솔을 엽니다.
7. New query 1(새 쿼리 1) 쿼리 창에 다음 SQL을 붙여넣습니다. *<database name>.<table name>*의 경우 복사한 SQL의 첫 번째 줄에서 데이터베이스 및 테이블 이름을 사용합니다.

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

8. 쿼리 실행을 선택합니다.
9. 드롭다운 메뉴에서 생성한 데이터베이스를 선택합니다.
10. New query 1(새 쿼리 1) 쿼리 창에 SQL 파일의 나머지 SQL을 붙여넣습니다.
11. 쿼리 실행을 선택합니다.

테이블 생성 이후 쿼리 실행 전에 파티션을 로드해야 합니다. 자세한 내용은 [보고서 파티션 업로드](#) 단원을 참조하십시오.

비용 및 사용 보고서

AWS는 하루에 여러 번 AWS CUR을 새로 고칩니다. Athena가 보고서를 새로 고치는 과정에서 AWS가 언제인지 알 수 있는 방법은 없습니다. 이로 인해 이전 데이터와 새 데이터를 조합하여 쿼리 결과가 나올 수 있습니다. 이를 완화하려면 비용 및 사용 보고서를 새로 고치는지 여부를 AWS 추적하는 데

이블을 생성하고 해당 테이블을 쿼리하여 AWS 가 데이터를 새로 고치는지 확인합니다. 이 테이블을 단 한 번만 생성해야 합니다. 그런 다음은 테이블을 최신 상태로 AWS 유지합니다.

새로 고침 테이블을 생성하려면

1. <https://console.aws.amazon.com/athena/>에서 Athena 콘솔을 엽니다.
2. New query 1(새 쿼리 1) 쿼리 창에 다음 SQL을 붙여넣습니다.

```
CREATE EXTERNAL TABLE IF NOT EXISTS cost_and_usage_data_status(
  status STRING)
ROW FORMAT SERDE
  'org.apache.hadoop.hive.q1.io.parquet.serde.ParquetHiveSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
)
LOCATION 's3://{S3_Bucket_Name}/{Report_Key}/cost_and_usage_data_status/'
```

3. 쿼리 실행을 선택합니다.

AWS 가 데이터를 새로 고치는지 확인하려면 Athena 콘솔을 사용하여 다음 SQL 쿼리를 실행합니다.

```
select status from cost_and_usage_data_status
```

보고서 파티션 업로드

비용 및 사용 보고서 데이터를 쿼리하려면 데이터를 Athena 테이블에 업로드해야 합니다. AWS 가 제공하는 각 새 AWS CUR 보고서에 대해 이 작업을 수행해야 합니다.

최신 파티션을 업로드하려면

1. <https://console.aws.amazon.com/athena/>에서 Athena 콘솔을 엽니다.
2. 테이블 이름 옆에 있는 세로로 된 점 세 개를 선택합니다.
3. 파티션 로드를 선택합니다.

파티션을 업로드하지 않은 경우 Athena에서 아무런 결과도 반환하지 않거나 데이터가 누락되었음을 나타내는 오류 메시지를 반환합니다.

Amazon Athena 쿼리 실행

데이터에 대해 Athena 쿼리를 실행하려면 먼저 Athena 콘솔을 사용하여 AWS 가 데이터를 새로 고치는지 확인한 다음 Athena 콘솔에서 쿼리를 실행합니다. SQL을 실행할 때 드롭다운 목록에서 올바른 데이터베이스를 선택했는지 확인합니다. 다음 SQL을 사용하여 상태를 확인할 수 있습니다.

```
select status from cost_and_usage_data_status
```

두 가지 가능한 결과는 READY 및 UPDATING입니다. 상태가 READY인 경우 Athena 데이터베이스를 쿼리할 수 있습니다. 상태가 UPDATING인 경우 Athena에서 완전하지 않은 결과를 반환할 수 있습니다.

AWS 가 데이터를 새로 고치고 있음을 확인한 후 자체 쿼리를 실행할 수 있습니다. 예를 들어 다음 쿼리는 mycostandusage_parquet이라는 예제 데이터베이스의 매월 서비스별 연간 누계를 표시합니다. 다음 쿼리는 2018년 연간 누계 비용을 보여줍니다. 연도를 업데이트하여 현재 연간 누계 비용을 확인합니다.

```
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;
```

열 이름

Athena 열 이름 제한은 비용 및 사용 보고서 열 이름 제한과 다릅니다. 즉, AWS CUR 데이터가 Athena 테이블에 업로드되면 열 이름이 변경됩니다. AWS 는 다음과 같이 변경합니다.

- 대문자 앞에 밑줄이 추가됨
- 대문자가 소문자로 교체됨
- 영숫자가 아닌 문자가 밑줄로 교체됨
- 중복 밑줄은 제거됨
- 선행 및 후행 밑줄이 제거됨
- 열 이름이 허용된 열 이름 길이보다 긴 경우 밑줄이 왼쪽에서 오른쪽 방향으로 제거됨

Note

이러한 규칙을 적용한 후 이름이 같은 열이 두 개 이상 있는 경우 일부 리소스 태그 열의 name. AWS merges 열이 중복됩니다.

예를 들어 열 이름 ExampleColumnName 및 Example Column Name은 각각 example_column_name 및 example_column_name이 됩니다.

보고서 데이터를 다른 리소스로 로드

비용 및 사용 보고서를 Amazon Redshift 및 Amazon Quick에 업로드하여 AWS 비용 및 사용량을 분석할 수 있습니다.

주제

- [Amazon Quick에 보고서 데이터 로드](#)
- [Amazon Redshift에 보고서 데이터 로드](#)

Amazon Quick에 보고서 데이터 로드

비용 및 사용 보고서를 Amazon Quick에 업로드할 수 있습니다.

빠른 업로드하는 방법에 대한 자세한 내용은 빠른 사용 설명서의 [Amazon S3 파일을 사용하여 데이터 세트 생성](#)을 참조하세요.

Amazon Redshift에 보고서 데이터 로드

이 섹션에서는 Amazon Redshift에 AWS CUR을 업로드하여 AWS 비용과 사용량을 분석하는 방법을 보여줍니다.

Important

Amazon Redshift 열은 대/소문자를 구분하지 않으며 사용자 정의 태그보다 엄격한 문자 제한이 적용됩니다. Amazon Redshift와 사용자 정의 태그 간의 충돌을 방지하기 위해 userTag1는 태그를 태그 userTag0, userTag2, 등으로 바 AWS 끕니다. Amazon Redshift 테이블을 생성하고 보고서를 여기에 업로드한 후 AWS정의된 태그를 사용자 정의 태그에 매핑하는 Amazon Redshift 테이블을 생성할 수 있습니다. 태그 테이블을 사용하면 원래 태그를 조회할 수 있습니다.

예를 들어 OWNER 및 Owner 태그가 있는 경우 Amazon Redshift에서는 'owner'라는 열이 두 개인 테이블을 생성할 수 없습니다. 대신 OWNER 및 Owner가 아닌 userTag0 및 userTag1 열로 보고서 테이블을 만든 다음 remappedUserTag 및 userTag 열로 테이블을 만듭니다. remappedUserTag 열은 AWS정의된 태그 userTag0 및를 저장userTag1하고 userTag 열은 원본 태그를 저장합니다. OWNER Owner

AWS 는 Amazon Redshift 테이블을 생성하고, 보고서를 업로드하고, 태그 테이블을 생성하고, 모든 태그 행을 태그 테이블에 삽입하는 명령을 제공합니다. 명령은 S3의 매니페스트 RedshiftCommands.sql 파일과 함께 저장된 파일과 Billing and Cost Management 콘솔의 Redshift 파일 헬퍼 파일에 제공됩니다. AWS 또한 RedshiftCommand 파일 업로드의 명령을 보고하는 것을 제어하는 RedshiftManifest 파일을 제공합니다. RedshiftCommand RedshiftManifest 파일을 삭제하거나 제거할 경우 RedshiftCommands 파일의 copy 명령이 중단됩니다.

결제 및 비용 관리 콘솔에서 **RedshiftCommands.sql** 파일 찾기

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. Amazon Redshift에 업로드할 보고서를 선택합니다.
4. 다음 서비스의 보고서 보기를 활성화했습니다. 옆의 Amazon Redshift를 선택합니다.
5. 대화 상자의 명령을 복사한 다음 SQL 클라이언트로 붙여 넣습니다.

다음 절차는 데이터베이스 및 Amazon Redshift에 친숙한 경우를 가정합니다.

Amazon Redshift에 비용 및 사용 보고서 업로드

1. Amazon Redshift 클러스터를 생성합니다.

자세한 정보는 Amazon Redshift 관리 가이드의 [클러스터 생성](#) 섹션을 참조하세요.

2. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
3. AWS CUR을 저장하는 Amazon S3 위치로 이동합니다.
4. RedshiftCommands.sql 파일을 엽니다.

파일에는 Amazon Redshift 테이블을 생성하고, Amazon S3에서 AWS CUR을 업로드하고, 사용자 정의 태그를 Amazon Redshift로 가져올 수 있는 태그 테이블을 생성하는 사용자 지정 명령이 포함되어 있습니다.

5. copy 명령에서 `<AWS_ROLE>`을 AWS CUR을 저장하는 Amazon S3 버킷에 액세스할 수 있는 권한이 있는 IAM 역할의 ARN으로 바꿉니다.
6. `<S3_BUCKET_REGION>`을 Amazon S3 버킷이 있는 리전으로 바꿉니다. 예를 들어 us-east-1입니다.
7. SQL 클라이언트를 사용하여 클러스터에 연결합니다.

자세한 정보는 Amazon Redshift 클러스터 가이드의 [Amazon Redshift 클러스터 및 데이터베이스 액세스](#)를 참조하세요.

8. RedshiftCommands.sql 파일의 SQL 명령을 SQL 클라이언트에 다음 순서로 복사합니다.
 - create table - 이 명령은 사용자 보고서에 맞게 사용자 정의된 스키마를 사용하는 Amazon Redshift 테이블을 만듭니다.
 - copy - 이 명령은 제공된 IAM 역할을 사용하여 S3에서 Amazon Redshift로 AWS CUR 파일을 업로드합니다.
 - create tag table - 이 명령은 AWS정의 태그를 사용자 정의 태그로 매핑할 수 있는 테이블을 만듭니다.
 - insert - 이 명령은 사용자 정의 태그를 태그 테이블에 삽입합니다.
9. AWS CUR의 모든 데이터를 Amazon Redshift로 복사한 후 SQL을 사용하여 데이터를 쿼리할 수 있습니다. Amazon Redshift의 데이터 쿼리에 대한 자세한 정보는 Amazon Redshift 데이터베이스 개발자 안내서의 [Amazon Redshift SQL](#)을 참조하세요.

Note

비용 및 사용 보고서의 열 수는 새로운 비용 할당 태그가 생성되거나 서비스에 새로운 제품 속성이 추가되는 경우 매월 변경될 수 있습니다. 매월 AWS CUR의 데이터를 새 테이블로 복사한 다음 관심 있는 열을 별도의 month-by-month 테이블로 복사하는 것이 좋습니다.

Billing Conductor를 사용하여 비용 및 사용량 보고서 구성

Billing Conductor에서 생성하는 각 결제 그룹에 대해 견적 AWS 비용 및 사용 보고서(AWS CUR)를 생성할 수 있습니다. 견적 AWS CUR은 표준 AWS CUR과 파일 형식, 세분화 및 열이 동일합니다. 견적서에는 특정 기간 동안 사용 가능한 가장 포괄적인 비용 및 사용량 데이터 집합이 포함됩니다. Billing Conductor에 대한 자세한 정보는 [Billing Conductor 사용 설명서](#)를 참조하세요.

주제

- [Billing Conductor AWS CUR과 표준 AWS CUR의 차이점 이해](#)
- [결제 그룹을 위한 견적 비용 및 사용 보고서 생성](#)

Billing Conductor AWS CUR과 표준 AWS CUR의 차이점 이해

Billing Conductor 구성을 사용하여 생성된 표준 비용 및 사용 보고서와 견적 AWS CUR에는 몇 가지 차이점이 있습니다.

- 표준 AWS CUR은 통합 결제 패밀리리의 각 계정에 대한 비용 및 사용량을 계산합니다. 청구 그룹당 견적 AWS CUR에는 계산 시 청구 그룹의 계정만 포함됩니다.
- 표준 AWS CUR은 인보이스 열을 한 번 채우고 인보이스가 생성됩니다 AWS. 견적 AWS CUR은 인보이스 열을 채우지 않습니다. 현재는 견적 결제 데이터를 AWS 기반으로에서 생성하거나 발급한 인보이스가 없습니다.

결제 그룹을 위한 견적 비용 및 사용 보고서 생성

다음 단계에 따라 결제 그룹에 대한 견적 AWS CUR을 생성합니다.

Note

레거시 AWS Cost and Usage Report 페이지는 결제 그룹 보기에 대한 보고서만 지원합니다. 결제 전송 보기에 대한 보고서를 생성하려면 데이터 내보내기 페이지를 사용합니다.

결제 그룹을 위한 견적 비용 및 사용 보고서 생성

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 보고서 테이블의 오른쪽 상단에서 설정을 선택합니다.
4. 견적 데이터 보기를 켭니다.
5. 활성화를 선택합니다.
6. 보고서 생성을 선택합니다.
7. 보고서 이름에 보고서 이름을 입력합니다.
8. 데이터 보기에서 견적을 선택합니다.
9. 결제 그룹을 선택합니다.

10. 추가 보고서 세부 정보에서 각 개별 리소스의 ID를 보고서에 포함하려면 리소스 ID 포함을 선택합니다.
11. 데이터 새로 고침 설정에서가 청구서를 완료한 후 계정에 환급, 크레딧 또는 지원 요금을 AWS 적용하는 경우 비용 및 사용 보고서를 새로 고칠지 여부를 선택합니다. 보고서가 새로 고침되면 새 보고서가 Amazon S3에 업로드됩니다.
12. 다음을 선택합니다.
13. S3 버킷에서 구성을 선택합니다.
14. S3 버킷 구성 대화 상자에서 다음 중 하나를 수행합니다.
 - 드롭다운 목록에서 기존 버킷을 선택하고 다음을 선택합니다.
 - 버킷 이름과 새 버킷을 생성할 AWS 리전을 입력하고 다음을 선택합니다.
15. 버킷 정책을 검토한 후 이 정책이 정확함을 확인함을 선택하고 저장을 선택합니다.
16. 보고서 경로 접두사에 보고서 이름의 접두어가 되는 보고서 경로 접두사를 입력합니다.

이 단계는 Amazon Redshift 또는 Quick의 경우 선택 사항이지만 Amazon Athena의 경우 필수입니다. 접두사를 지정하지 않을 경우 기본 접두사는 7단계에서 보고서에 지정한 이름과 보고서 날짜 범위이고 형식은 다음과 같습니다. /report-name/date-range/
17. 시간 세부 수준에 대해 다음 중 하나를 선택합니다.
 - 시간별: 보고서의 항목을 시간별로 집계하려면 선택합니다.
 - 일별: 보고서의 항목을 일별로 집계하려면 선택합니다.
18. 보고서 버전 관리에서, 보고서의 각 버전을 이전 버전을 덮어쓸지 또는 이전 버전과 별도로 추가 제공할지를 선택합니다.

보고서를 덮어쓰면 Amazon S3 스토리지 비용을 절약할 수 있습니다. 새 보고서 버전을 제공하면 시간이 지남에 따라 결제 데이터의 감사 가능성을 개선할 수 있습니다.
19. 보고서 데이터 통합에서 비용 및 사용 보고서를 Amazon Athena, Amazon Redshift 또는 Quick에 업로드할지 여부를 선택합니다. 보고서는 다음 형식으로 압축됩니다.
 - Athena: parquet 형식
 - Amazon Redshift 또는 Quick: .gz 압축
20. 다음을 선택합니다.
21. 보고서 설정을 검토했으면 [Review and Complete]를 선택합니다.

데이터 사전

비용 및 사용 보고서에는 사용에 대한 세부 정보가 포함됩니다. 다음 섹션에서는 보고서에 표시되는 열의 하위 집합 및 해당 정의를 나열하고 설명합니다.

AWS 비용 및 사용 보고서(AWS CUR)에 표시될 수 있는 열의 전체 목록과 열이 적용되는 서비스를 다운로드하려면 [Column_Attribute_Service.zip](#)을 다운로드합니다. 이 심표로 구분된 값(CSV) 목록에는 Identity, Bill, Lineltem, Reservation, Pricing, Product 열이 포함됩니다.

모든 비용 및 사용 보고서에는 Identity, Bill, Lineltem 열이 포함됩니다. 다른 모든 열은 월별 AWS 사용량이 해당 열을 채우는 데이터를 생성하는 경우에만 보고서에 포함됩니다.

주제

- [자격 증명 세부 정보](#)
- [결제 세부 정보](#)
- [항목 세부 정보](#)
- [예약 세부 정보](#)
- [요금 세부 정보](#)
- [제품 세부 정보](#)
- [리소스 태그 세부 정보](#)
- [절감형 플랜 세부 정보](#)
- [비용 범주 세부 정보](#)
- [할인 세부 정보](#)
- [분할 항목 세부 정보](#)

자격 증명 세부 정보

AWS 비용 및 사용 보고서의 identity 헤더 아래에 있는 열은 모든 비용 및 사용 보고서에 표시되는 정적 필드입니다.

보고서의 자격 증명 행 항목을 사용하여 여러 AWS CUR 파일로 분할된 특정 행 항목을 찾을 수 있습니다. 여기에는 다음 열이 포함됩니다.

identity/LineItemId

- 설명: 이 필드는 각 항목에 대해 생성되며 지정된 파티션마다 고유합니다. 이렇게 AWS 해도 필드가 CUR의 전체 전송(즉, 업데이트의 모든 파티션)에서 고유하다는 보장은 없습니다. 항목 ID는 각기 다른 비용 및 사용 보고서 사이에서 일관적이지 않으며, 각기 다른 보고서에 걸쳐 동일한 항목을 식별하는 데 사용할 수 없습니다.
- 예제: 11월 29일에 대해 생성된 보고서는 여러 파일이 필요할 정도로 클 수 있습니다. LineItemId는 11월 29일 AWS CUR 파일 간에 일관성이 있지만 11월 30일 보고서의 동일한 리소스에 대한 LineItemId와는 일치하지 않습니다.

identity/TimeInterval

- 설명: 이 항목에 적용되는 시간 간격으로, YYYY-MM-DDTHH:mm:ssZ/YYYY-MM-DDTHH:mm:ssZ 형식을 갖습니다. 시간 간격은 UTC이고, 보고서의 세부 수준에 따라 일별 또는 시간별이 될 수 있습니다.
- 예제: TimeInterval 2017-11-01T00:00:00Z/2017-12-01T00:00:00Z에는 2017년 11월의 전체 달이 포함됩니다.

결제 세부 정보

AWS 비용 및 사용 보고서의 bill 헤더 아래에 있는 열은 모든 비용 및 사용 보고서에 표시되는 정적 필드입니다. 보고서의 청구서 항목을 사용하여 청구 유형 및 청구 기간의 시작일과 종료일 같은 보고서에 적용되는 구체적인 청구 관련 세부 정보를 찾아볼 수 있습니다. 여기에는 다음 열이 포함됩니다.

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

인보이스 또는 트랜잭션이 다른 AWS 서비스를 위한 것인지 AWS Marketplace 아니면 다른 서비스를 구매하기 위한 것인지 식별하는 데 도움이 됩니다. 가능한 값은 다음과 같습니다.

- AWS - AWS Marketplace내부 이외의 AWS 서비스에 대한 거래를 식별합니다.
- AWS Marketplace - AWS Marketplace에서의 구매를 식별합니다.

bill/BillingPeriodEndDate

이 보고서의 청구 기간 종료일(UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ입니다.

bill/BillingPeriodStartDate

이 보고서의 청구 기간 시작일(UTC)입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ입니다.

bill/BillType

이 보고서가 다루는 결제 유형입니다. 3가지 청구 유형이 있습니다.

- 연간 - 한 달 동안 사용한 서비스의 항목
- 구매 - 선결제 서비스 요금의 항목
- 환급 - 환급의 항목

I

bill/InvoiceId

특정 항목과 연결된 ID입니다. 보고서가 완성될 때까지 InvoiceId는 비어 있습니다.

bill/InvoicingEntity

인보이스를 발행하는 AWS 엔터티입니다. 가능한 값은 다음과 같습니다.

- Amazon Web Services, Inc. - 해당하는 경우 전 세계 고객에게 청구서를 발행하는 기관입니다.
- Amazon Web Services India Private Limited - 인도에 기반을 둔 고객에게 청구서를 발행하는 기관입니다.
- Amazon Web Services South Africa Proprietary Limited - 남아프리카 고객에게 청구서를 발행하는 기관입니다.

P

bill/PayerAccountId

지급 계정의 계정 ID입니다. 의 조직의 경우 관리 계정의 계정 ID AWS Organizations입니다.

항목 세부 정보

AWS 비용 및 사용 보고서의 `lineItem` 헤더 아래에 있는 열은 모든 비용 및 사용 보고서에 표시되는 정적 필드입니다. 사용량에 대한 모든 비용 및 사용량 정보를 다룹니다. 여기에는 다음 열이 포함됩니다.

[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`

이 항목을 호스팅하는 가용 영역입니다. 예: `us-east-1a` 또는 `us-east-1b`.

B

`lineItem/BlendedCost`

`BlendedRate`에 `UsageAmount`를 곱한 값과 같습니다.

Note

`LineItemType`이 할인인 항목의 경우 `BlendedCost`는 비어 있습니다. 할인은 멤버 계정의 일반 비용만 사용하여 계산되며 멤버 계정 및 SKU별로 집계됩니다. 따라서 `BlendedCost`는 할인에 사용할 수 없습니다.

`lineItem/BlendedRate`

`BlendedRate`는 조직 전체의 각 SKU에 대해 발생하는 평균 비용을 나타냅니다.

예를 들어 Amazon S3 혼합 요금은 월별로 저장된 데이터의 양으로 분할된 스토리지의 총 비용입니다. RI가 포함된 계정의 경우 혼합 요금은 RI 및 온디맨드 인스턴스의 평균 비용으로 계산됩니다.

혼합 요금은 관리 계정 수준에서 계산되며 각 멤버 계정에 비용을 할당하는 데 사용됩니다. 자세한 내용은 AWS Billing 사용 설명서의 [혼합 요금 및 비용](#)을 참조하세요.

C

`lineItem/CurrencyCode`

이 항목에 표시된 통화입니다. 모든 AWS 고객에게는 기본적으로 미국 달러로 요금이 청구됩니다. 청구 통화를 변경하려면 AWS Billing 사용 설명서의 [청구서 결제에 사용하는 통화 변경](#)을 참조하세요.

L

lineltem/LegalEntity

특정 제품 또는 서비스의 등록 판매자입니다. 대부분의 경우 인보이스 발행 주체와 법인은 동일합니다. 타사 AWS Marketplace 트랜잭션의 경우 값이 다를 수 있습니다. 가능한 값은 다음과 같습니다.

- Amazon Web Services, Inc. - AWS 서비스를 판매하는 엔터티입니다.
- Amazon Web Services India Private Limited - 인도에서 AWS 서비스의 리셀러 역할을 하는 인도 현지 기업입니다.

lineltem/LineltemDescription

항목 유형의 설명입니다. 예를 들어 사용량 항목의 설명은 지정된 기간 동안 어떤 유형의 사용량이 발생했는지 요약합니다.

유연한 크기의 RI의 경우 설명은 그 혜택이 적용되는 RI와 일치합니다. 예를 들어 항목이 t2.micro에 해당하고 t2.small RI가 사용량에 적용된 경우 lineltem/LineltemDescription에 t2.small이 표시됩니다.

RI 할인이 포함된 사용량 항목의 설명에는 항목에 포함되는 요금제를 포함합니다.

lineltem/LineltemType

이 항목에 적용되는 요금 유형입니다. 가능한 유형은 다음과 같습니다.

- BundledDiscount - 다른 서비스 또는 기능의 사용에 따라 서비스 또는 기능을 무료 또는 할인된 가격으로 사용할 수 있는 사용량 기반 할인입니다.
- Credit - 청구서에 AWS 적용된 모든 크레딧입니다. 세부 정보는 설명 열을 참조하십시오. AWS가 청구서 금액이 확정된 후 해당 월에 대해 계정에 크레딧을 적용하는 경우에는 AWS가 보고서를 완성된 이후에 업데이트할 수 있습니다.
- Discount - 사용량에 AWS 적용된 모든 할인. 이 특정 항목의 이름은 다를 수 있으며 할인을 기준으로 구문 분석해야 합니다. 자세한 내용은 lineltem/LineltemDescription 열을 참조하세요.
- DiscountedUsage - 예약 인스턴스(RI) 혜택을 보유한 모든 인스턴스의 요금입니다.
- Fee - 구독에 대해 지불하는 모든 선결제 연간 요금입니다. 예를 들어 전체 선결제 RI 또는 부분 선결제 RI에 대해 지불하는 선결제 요금입니다.
- Refund - AWS가 환급한 음수 값의 요금입니다. 청구서를 완료한 후 해당 월의 계정에 환급을 AWS 적용하는 경우 완료 후 설명 열에서 details.AWS might 업데이트 보고서를 확인합니다.

- **RIFee** - 구독에 대해 매달 발생하는 요금입니다. 예를 들면, 부분 선결제 RI, 선결제 없음 RI 및 전체 선결제에 대해 매달 지불하는 기본 요금입니다. 모든 선결제 예약의 경우 RIFee 가 0 USD일 수 있지만, 해당 예약 유형에 대해서는 여전히 이 행이 채워져 reservation/AmortizedUpfrontFeeForBillingPeriod 및 reservation/ReservationARN 등의 다른 열을 제공합니다.
- **Tax** - 청구서에 AWS 적용된 모든 세금입니다. 예: VAT 또는 미국 판매세.
- **Usage** - 온디맨드 인스턴스 요금이 부과되는 모든 사용량입니다.
- **SavingsPlanUpfrontFee** - 전체 선결제 또는 부분 선결제 절감형 플랜 구매 시 발생하는 일회성 선결제 요금입니다.
- **SavingsPlanRecurringFee** - 선결제 없음 또는 부분 선결제 절감형 플랜에 해당하는 모든 시간당 기본 요금입니다. 절감형 플랜 기본 요금은 선결제 없음 또는 부분 선결제 절감형 플랜을 구매한 날에 청구서에 처음으로 가산됩니다. 최초 구매 후는 이후 각 결제 기간의 첫날에 반복 요금을 AWS 추가합니다.
- **SavingsPlanCoveredUsage** - 절감형 플랜이 적용되는 모든 온디맨드 비용입니다. 절감형 플랜 적용 사용량 항목은 상응하는 절감형 플랜 부정 항목에 의해 상쇄됩니다.
- **SavingsPlanNegation** - 해당 절감형 플랜 보장 사용 항목과 관련된 절감형 플랜 혜택을 통한 모든 상쇄 비용입니다.

절감형 플랜 항목에 대한 자세한 정보와 지침은 [절감형 플랜 이해](#) 섹션을 참조하세요.

N

lineItem/NetUnblendedCost

항목에 대해 지불하는 실제 할인 후 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

lineItem/NetUnblendedRate

항목에 대해 지불하는 실제 할인 후 요금입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

lineItem/NormalizationFactor

인스턴스가 테넌시를 공유하는 한는 인스턴스 패밀리 및 AWS 리전의 모든 인스턴스 크기에 모든 리전 Linux 또는 Unix Amazon EC2 및 Amazon RDS RI 할인을 적용할 AWS 수 있습니다. 또한 조직의 멤버 계정에 대해 RI 할인을 적용합니다. 모든 신규 및 기존 Amazon EC2 및 Amazon RDS 유연한 크기 RI 는 인스턴스 크기를 기반으로 하는 정규화 인자에 따라 크기 조정됩니다. 다음 표에는 각 인스턴스 크기에 AWS 적용되는 정규화 인자가 나와 있습니다.

Amazon EC2 유연한 크기 RI에 대한 정규화 인자

인스턴스 크기	정규화 인자
nano	0.25
micro	0.5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
4xlarge	32
8xlarge	64
10xlarge	80
16xlarge	128
32xlarge	256

lineItem/NormalizedUsageAmount

유연한 크기의 RI에 대해 정규화 유닛에서 발생한 사용량입니다. NormalizedUsageAmount는 UsageAmount에 NormalizationFactor를 곱한 값과 같습니다.

O

lineItem/Operation

이 행 항목에서 다루는 특정 AWS 작업입니다. 이 항목의 구체적인 사용량에 대해 설명합니다. 예를 들어 RunInstances 값은 Amazon EC2 인스턴스 작업을 나타냅니다.

P

lineItem/ProductCode

측정된 제품의 코드입니다. 예를 들어 Amazon EC2는 Amazon Elastic Compute Cloud의 제품 코드입니다.

R

lineItem/ResourceId

(선택 사항) 보고서에 개별 리소스 ID를 포함하기로 선택했다면, 이 열은 프로비저닝한 리소스 ID를 포함합니다. 예를 들어 Amazon S3 스토리지 버킷, Amazon EC2 컴퓨팅 인스턴스 또는 Amazon RDS 데이터베이스 각각에 리소스 ID가 있습니다. 데이터 전송 및 API 요청과 같이 인스턴스화된 호스트와 연결되지 않은 사용량 유형, 그리고 할인, 크레딧 및 세금과 같은 항목 유형에 대해서는 이 필드가 빈 칸입니다. 다음 표에는 일반적인 AWS 서비스의 리소스 식별자 목록이 나와 있습니다.

AWS 리소스 식별자

AWS 서비스	리소스 식별자
Amazon CloudFront	배포 ID
Amazon CloudSearch	검색 도메인
Amazon DynamoDB	DynamoDB 테이블
Amazon Elastic Compute Cloud - Amazon EBS	Amazon EBS 볼륨
- Amazon Elastic Compute Cloud	인스턴스 ID

AWS 서비스	리소스 식별자
Amazon Elastic Compute Cloud - CloudWatch	인스턴스 ID에 대한 CloudWatch 요금
Amazon EMR	MapReduce 클러스터
Amazon ElastiCache	캐시 클러스터
Amazon OpenSearch Service	검색 도메인
Amazon Glacier	볼트
Amazon Relational Database Service	Database
Amazon Redshift	Amazon Redshift 클러스터
Amazon Simple Storage Service	Amazon S3 버킷
Amazon Virtual Private Cloud	VPN ID
AWS Lambda	Lambda 함수 이름

T

lineItem/TaxType

이 항목에 AWS 적용된 세금 유형입니다.

U

lineItem/UnblendedCost

UnblendedCost는 UnblendedRate에 UsageAmount를 곱한 값입니다.

lineItem/UnblendedRate

AWS Organizations를 사용하는 계정에 대한 통합 결제에서 일반 요금은 개별 계정의 서비스 사용과 관련된 요금입니다.

RI 할인이 적용된 Amazon EC2 및 Amazon RDS 항목의 경우 UnblendedRate는 0입니다. RI 할인이 있는 항목의 LineItemType는 DiscountedUsage입니다.

lineItem/UsageAccountId

이 항목에서 사용된 계정의 계정 ID입니다. 조직의 경우, 이는 관리 계정일 수도 있고 멤버 계정일 수도 있습니다. 이 필드를 사용하여 계정별 비용과 사용량을 추적할 수 있습니다.

lineItem/UsageAmount

지정된 기간 동안 발생한 사용량입니다. 크기가 유연한 예약 인스턴스의 경우 reservation/TotalReservedUnits 열을 대신 사용합니다.

Note

특정 구독 요금의 UsageAmount는 0입니다.

lineItem/UsageEndDate

해당 항목의 종료 날짜와 시간(UTC)입니다. 배타적입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ입니다.

lineItem/UsageStartDate

이 항목의 시작 날짜와 시간(UTC)입니다. 포괄적입니다. 형식은 YYYY-MM-DDTHH:mm:ssZ입니다.

lineItem/UsageType

이 항목의 사용량 세부 정보입니다. 예를 들어 USW2-BoxUsage:m2.2xlarge는 미국 서부(오레곤) 리전에 있는 M2 고용량 메모리 더블 엑스트라 라지 인스턴스를 설명합니다.

예약 세부 정보

AWS 비용 및 사용 보고서의 reservation 헤더 아래에 있는 열은 예약된 리소스에 대한 세부 정보를 제공합니다.

[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

- 설명: 사용 시간 동안 분할 상환되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다. 값은 다음과 같습니다. $RIAmortizedUpfrontFeeForBillingPeriod * \frac{\text{The normalized usage amount for DiscountedUsage line items}}{\text{The normalized usage amount for the RIFee}}$. 선결제 없음 RI의 경우 선결제 금액이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.
- 해당하는 항목: DiscountedUsage
- 샘플 값: 0.05, 0.17, 0.15
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/AmortizedUpfrontFeeForBillingPeriod

- 설명: 이 예약의 선결제 요금이 이 결제 기간 동안 얼마나 부과되는지를 알려줍니다. 이 달 동안 분할 상환되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다. 선결제 없음 RI의 경우 선결제 요금이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.
- 해당하는 항목: RIFee
- 샘플 값: 29.15, 200.67, 214.43
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache

- Amazon RDS

reservation/AvailabilityZone

- 설명: 이 항목과 연결된 리소스의 가용 영역입니다.
- 해당하는 항목: Fee, Refund, RIFee
- 샘플 값: us-east-1, us-east-1b, eu-west-1b, ap-southeast-2a
- 서비스:
 - Amazon EC2

E

reservation/EffectiveCost

- 설명: 유효한 시간당 요금으로 평균을 낸 RI의 선결제 및 시간당 요금의 합입니다. EffectiveCost는 amortizedUpfrontCostForUsage를 가져온 다음 recurringFeeForUsage에 더하여 계산됩니다. 자세한 내용은 [Amazon EC2 예약 인스턴스 요금](#)을 참조하세요.
- 해당하는 항목: DiscountedUsage
- 샘플 값: 0.23, 0.68, 0.10
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon ElastiCache
 - Amazon RDS

reservation/EndTime

- 설명: 연결된 RI 리즈 기간의 마지막 날짜입니다.
- 해당하는 항목: RIFee
- 샘플 값: 2019-05-15T04:23:14.000Z, 2020-02-08T17:32:15.000Z, 2019-07-14T00:00:33.000Z

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

M

reservation/ModificationStatus

- 설명: RI 리즈가 수정되었는지 변경된 사항이 없는지 보여줍니다.
 - 원본: 구입한 RI가 변경된 적이 없습니다.
 - 시스템: 구입한 RI가 콘솔 또는 API를 사용하여 변경되었습니다.
 - 수동: 구매한 RI가 AWS Support 지원을 사용하여 수정되었습니다.
 - ManualWithData: 구매한 RI는 AWS Support 지원을 통해 수정되었으며 RI에 대한 추정치를 AWS 계산했습니다.
- 해당하는 항목: RIFee
- 샘플 값: Original, System, Manual, ManualWithData
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

N

reservation/NetAmortizedUpfrontCostForUsage

사용 시간 동안 분할 상환되는 전체 선결제 RI 및 부분 선결제 RI의 초기 선결제 금액입니다(해당하는 경우) 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetAmortizedUpfrontFeeForBillingPeriod

청구 기간 동안의 예약 선결제 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetEffectiveCost

유효한 시간당 요금으로 평균을 낸 RI의 선결제 및 시간당 요금의 합입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetRecurringFeeForUsage

기본 사용 요금의 할인 후 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetUnusedAmortizedUpfrontFeeForBillingPeriod

청구 기간의 미사용 분할 상환 순 선결제 요금입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetUnusedRecurringFee

할인 후 부분 선결제 및 선결제 없음 RI에 대해 미사용 예약 시간과 연결된 기본 요금입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NetUpfrontValue

할인이 적용된 RI의 선결제 금액입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

reservation/NormalizedUnitsPerReservation

- 설명: 예약 구독의 각 인스턴스에 대한 정규화 단위의 수입입니다.
- 해당하는 항목: RIFee
- 샘플 값: 1316, 54.5, 319
- 서비스:
 - Amazon RDS

reservation/NumberOfReservations

- 설명: 이 구독권이 적용되는 예약의 수입입니다. 예를 들어 RI 구독권 하나에 연관 RI 예약 4개가 포함될 수 있습니다.

- 해당하는 항목: Fee, RIFee, Refund, Credit
- 샘플 값: 5, 50, 500
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

R

reservation/RecurringFeeForUsage

- 설명: 부분 선결제 RI 및 선결제 없음 RI의 경우, 사용 시간 동안 분할 상환되는 기본 요금입니다. 값은 다음과 같습니다. $\frac{\text{The unblended cost of the RIFee} * \text{The sum of the normalized usage amount of Usage line items}}{\text{The normalized usage amount of the RIFee for size flexible Reserved Instances}}$. 전체 선결제 RI의 기본 요금 금액이 0보다 크지 않으므로 전체 선결제 RI의 값은 0입니다.
- 해당하는 항목: DiscountedUsage
- 샘플 값: 0.139, 0.729, 0.018
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon ElastiCache
 - Amazon RDS

reservation/ReservationARN

- 설명: 이 항목이 혜택을 받는 RI의 Amazon 리소스 이름(ARN)입니다. 이는 또한 'RI 리즈 ID'라고 합니다. 이는 이 특정 AWS 예약 인스턴스의 고유 식별자입니다. 값 스트링에는 AWS 서비스 이름과 RI를 구입한 리전이 포함되어 있습니다.

- 해당하는 항목: Fee, RIFee, DiscountedUsage, Refund, Credit
- 샘플 값: 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
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

S

reservation/StartTime

- 설명: 연결된 예약 인스턴스 기간의 시작 날짜입니다.
- 해당하는 항목: RIFee
- 샘플 값: 2018-07-29T02:56:10.000Z, 2017-08-21T15:58:47.000Z, 2019-02-01T22:01:34.000Z
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/SubscriptionId

- 설명: 항목과 관련 제안을 매핑하는 고유한 식별자입니다. RI ARN을 AWS 예약 인스턴스의 식별자로 사용하는 것이 좋지만 둘 다 사용할 수 있습니다.
- 해당하는 항목: Fee, RIFee, DiscountedUsage, Refund, Credit 및 Usage
- 샘플 값: 123456789, 111122222, 333344444

- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

T

reservation/TotalReservedNormalizedUnits

- 설명: 예약 구독의 모든 인스턴스에 대한 총 예약 정규화 단위 수입입니다. 예를 곱하여 총 정규화 단위 `reservation/NormalizedUnitsPerReservation`를 AWS 계산합니다 `reservation/NumberOfReservations`.
- 해당하는 항목: DiscountedUsage
- 샘플 값: 40320, 3647.99, 17928.77
- 서비스:
 - Amazon EC2
 - Amazon RDS

reservation/TotalReservedUnits

- 설명: TotalReservedUnits은 요금 및 RI 요금 항목을 둘 다 채웁니다.
 - 요금 항목: 전체 기간에 대한 구독에서 구매한 리즈의 전체 양에 대해 예약된 유닛의 총 숫자입니다.
- 이 값은 NumberOfReservations에 UnitsPerReservation 값을 곱하여 계산됩니다. 예를 들면 5 RI x 한 달 744시간 x 12개월은 44,640입니다.
- RIFee 항목(매달 기본 비용): 특정 RI 구독에서 Amazon EC2 시간의 총 수와 같은 구독에서 이용 가능한 유닛의 총 수입입니다.

예를 들면 5 RI x 744시간은 3,720입니다.

- 해당하는 항목: Fee, RIFee, Refund, Credit

- 샘플 값: 26208, 98.19, 15796
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

U

reservation/UnitsPerReservation

- 설명: UnitsPerReservation은 요금 및 RI 요금 항목을 둘 다 채웁니다.
 - 요금 항목: 구독 기간에 구매한 RI 시간의 전체 숫자 등 구독에 대해 예약된 유닛의 총 숫자입니다.

예를 들면 한 달 744시간 x 12개월은 총 8,928 시간/유닛입니다.
 - RIFee 항목(매달 기본 비용): 특정 RI 구독에서 Amazon EC2 시간의 총 수와 같은 구독에서 이용 가능한 유닛의 총 수입니다.

예를 들면, 1 유닛 x 744시간은 744입니다.
- 해당하는 항목: Fee, RIFee, Refund, Credit
- 샘플 값: 334.0, 486.72, 18455
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon DynamoDB
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/UnusedAmortizedUpfrontFeeForBillingPeriod

- 설명: 전체 선결제 RI 및 부분 선결제 RI에 대한 초기 선결제 요금의 분할 상환 부분입니다. 선결제 없음 RI의 경우 선결제 금액이 없으므로 선결제 없음 RI의 값은 0입니다. 현재 전용 호스트 예약에는 이 값을 제공하지 않습니다. 향후 업데이트에서 변경이 이루어질 예정입니다.
- 해당하는 항목: RIFee
- 샘플 값: 6.05, 1.97, 0.17
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/UnusedNormalizedUnitQuantity

- 설명: 유연한 크기의 리전별 RI에 대해 결제 기간 동안 사용하지 않은 미사용 정규화 단위 수입니다.
- 해당하는 항목: RIFee
- 샘플 값: 25.00, 3.50, 274.33
- 서비스:
 - Amazon RDS

reservation/UnusedQuantity

- 설명: 이 결제 기간 동안 사용하지 않은 RI 시간의 수입니다.
- 해당하는 항목: RIFee line item
- 샘플 값: 209.65110408, 191.00000000, 176.00000000
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/UnusedRecurringFee

- 설명: 부분 선결제 및 선결제 없음 RI에 대해 미사용 예약 시간과 연결된 기본 요금입니다. 전체 선결제 RI의 기본 요금이 0보다 크지 않으므로 전체 선결제 RI의 값은 0입니다.
- 해당하는 항목: RIFee
- 샘플 값: 0.02971114, 0.19190000, 1.37280000
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

reservation/UpfrontValue

- 설명: AWS 예약 인스턴스에 대해 지불된 선결제 가격입니다. 선결제 없음 RI의 경우, 이 값은 0입니다.
- 해당하는 항목: RIFee
- 샘플 값: 150.00, 1000.00, 2000.00
- 서비스:
 - Amazon EC2
 - OpenSearch Service
 - Amazon Redshift
 - Amazon ElastiCache
 - Amazon RDS

요금 세부 정보

AWS 비용 및 사용 보고서의 pricing 헤더 아래에 있는 열에는 품목에 대한 가격이 포함됩니다. 요금 열은 AWS Price List Service API를 기반으로 합니다. AWS Price List Service API에는 스팟 인스턴스,의 제품 AWS Marketplace, 선결제 연간 구독 요금(Fee) 및 월별 기본 요금()이 포함되지 않습니다RIFee. 이러한 열에는 다음이 포함되지만 이에 제한되지는 않습니다.

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

RI가 예약되어 있는 기간입니다.

O

pricing/OfferingClass

- 설명: 예약형 인스턴스에서 제공하는 클래스를 설명합니다. 예약 인스턴스를 구입할 경우 표준 또는 전환형 제공 클래스 중에서 선택할 수 있습니다.
- 샘플 값: Standard, Convertible
- 서비스:
 - Amazon DynamoDB
 - Amazon EC2
 - Amazon ElastiCache
 - OpenSearch Service
 - Amazon RDS
 - Amazon Redshift

P

pricing/publicOnDemandCost

퍼블릭 온디맨드 인스턴스 요금 기반의 항목의 총 비용. 여러 온디맨드 퍼블릭 요금이 있는 SKU를 보유하고 있는 경우 가장 높은 티어에 상응하는 요금이 표시됩니다. 예를 들어 프리 티어 또는 계층화된 요금을 제공하는 서비스가 있습니다.

pricing/publicOnDemandRate

특정 사용 항목에 대한 이 결제 기간 내의 퍼블릭 온디맨드 인스턴스 요금. 여러 온디맨드 퍼블릭 요금이 있는 SKU를 보유하고 있는 경우 가장 높은 티어에 상응하는 요금이 표시됩니다. 예를 들어 프리 티어 또는 계층화된 요금을 제공하는 서비스가 있습니다.

pricing/PurchaseOption

어떻게 이 항목에 지불할 결정을 했습니까. 유효한 값은 All Upfront, Partial Upfront, No Upfront입니다.

R

pricing/RateCode

제품/제안/요금 티어 조합의 고유 코드입니다. 제품 및 조건 조합은 요금 차원이 여러 개 있을 수 있습니다(예: 저사용 티어, 고사용 티어 등).

pricing/RateId

항목에 대한 요금의 ID입니다.

T

pricing/term

AWS 사용량이 예약인지 온디맨드인지 여부입니다.

U

pricing/unit

가 사용 비용을 계산하는 데 AWS 사용한 요금 단위입니다. 예를 들어 Amazon EC2 인스턴스 사용량의 요금 단위는 시간입니다.

제품 세부 정보

product 열은 비용을 초래한 제품 및 항목에 대한 메타데이터를 제공합니다. 제품 열은 동적이며 비용 및 사용 보고서의 표시 여부는 결제 기간 내의 제품 사용에 따라 달라집니다. 요금 열은 AWS Price List Service API를 기반으로 합니다. AWS Price List Service API에는 스팟 인스턴스, AWS Marketplace의 제품, 선결제 연간 구독 요금(Fee) 및 월별 기본 요금()이 포함되지 않습니다RIFee.

[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

- 설명: DevOps Guru 서비스가 DevOps Guru 서비스를 활성화하는 데 사용하는 API 수를 설명합니다.
- 샘플 값: 100, 500, 10000
- 서비스:
 - Amazon DevOps Guru

product/attachmentType

- 설명: Transit Gateway 또는 클라우드 WAN 서비스에 대한 연결 유형을 설명합니다.
- 샘플 값: VPC, AWS Site-to-Site VPN, AWS DirectConnect, Connect, Transit Gateway
- 서비스:
 - Amazon Virtual Private Cloud
 - AWS 클라우드 WAN

product/availability

- 설명: 다양한 AWS 스토리지 옵션의 가용성을 설명합니다.
- 샘플 값: 99.99%, 99.5%
- 서비스:
 - Amazon Glacier
 - Amazon S3
 - AWS Elemental MediaStore
 - AWS RoboMaker

C

product/cacheType

- 설명: 고객이 자주 읽는 데이터의 성능을 향상시키기 위해 HDD 기반 파일 시스템에서 읽기 전용 SSD 캐시를 선택한 조항에 대해 설명합니다.

예를 들어, RC20은 파일 시스템 HDD 스토리지 용량의 20%로 자동 크기가 조정되는 읽기 전용 SSD 캐시가 있음을 나타냅니다.

- 샘플 값: RC20, N/A
- 서비스:
 - Amazon FSx

product/capacitystatus

- 설명: 용량 예약 상태를 설명합니다.
- 샘플 값: UnusedCapacityReservation, AllocatedCapacityReservation, Used

- 서비스:
 - Amazon EC2

product/clockspeed

- 설명: AWS 인스턴스의 작동 속도를 설명합니다.
- 샘플 값: 2.4 GHz, 2.6 GHz
- 서비스:
 - Amazon DocumentDB
 - Amazon EC2
 - Amazon MQ
 - Amazon Neptune
 - Amazon RDS
 - AWS 데이터베이스 마이그레이션 서비스

product/component

- 설명: SageMaker AI의 기능에 매핑됩니다.

예를 들어, 사용자가 SageMaker AI 노트북을 실행 중인 경우 제품은 노트북이라는 구성 요소 속성을 갖게 됩니다. 사용자가 추론을 위해 모델을 배포하고 호스팅한 경우 구성 요소 속성이 호스팅인 제품이 표시됩니다.

- 샘플 값: Notebook, Hosting
- 서비스:
 - Amazon SageMaker AI

D

product/databaseedition

- 설명: 다양한 개발, 배포 시나리오 및 특정 애플리케이션 목적에 적합한 데이터베이스 소프트웨어를 설명합니다.
- 서비스:
 - Amazon RDS

product/dataTransfer

- 설명: AWS 데이터가에서 AWS 인터넷으로 이동되거나 해당 리전 또는 가용 영역의 AWS 인스턴스 간에 이동할 때마다 데이터 전송이 발생합니다. 리전 간 및 가용 영역 간 데이터 전송에는 GB당 계산된 비용이 발생합니다.
- 서비스:
 - AWS Systems Manager

product/dedicatedEbsThroughput

- 설명: 인스턴스 간(예: Amazon EC2 인스턴스와 Amazon EBS 볼륨)에 전용 처리량을 설명하며, 사용하는 인스턴스 유형에 따라 초당 500~10,000Mbit(Mbps) 범위에서 처리량을 선택할 수 있습니다. 전용 처리량이 EC2 인스턴스에서 Amazon EBS I/O와 기타 트래픽 간의 경합을 최소화하여 Amazon EBS 볼륨의 성능이 극대화됩니다.
- 샘플 값: 200 Mbps, Upto 5000 Mbps
- 서비스:
 - Amazon EC2
 - Amazon Neptune
 - Amazon RDS

product/deploymentoption

- 설명: 환경 인프라의 위치를 설명합니다. AWS 클라우드 배포 모델은 public, on-premise, hybrid입니다.
- 샘플 값: Multi-AZ, Single-AZ
- 서비스:
 - Amazon MQ
 - Amazon Neptune
 - Amazon RDS

product/description

- 설명: 특정 AWS 서비스에 대한 설명입니다.
- 서비스:

- AWS CodePipeline
- AWS Device Farm
- AWS Elemental MediaConvert
- AWS Elemental MediaStore

product/destinationCountryISOCode

- 설명: SMS가 전송된 대상 국가 ISO 3166-1 알파-2 코드를 설명합니다.

자세한 내용은 https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2를 참조하세요.

- 샘플 값: 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
- 서비스:
 - Amazon Simple Notification Service

product/directconnectlocation

- 설명: 고객에서 AWS 로의 개인 전용 네트워크 연결이 있는 위치를 지정합니다.
- 샘플 값: Equinix DC1 - DC6, Equinix DC10 - DC11, Global Switch Singapore
- 서비스:
 - Direct Connect

product/directorysize

- 설명: 디렉터리 또는 폴더의 메타 정보를 저장하는 데 사용되는 디스크 공간입니다.
- 서비스:
 - Direct Connect

product/directorytype

- 설명: 디렉터리가 파일인지 다른 디렉터리인지를 지정합니다.
- 서비스:
 - Direct Connect

product/directorytypedescription

- 설명: 디렉터리에 지정된 의미 있는 이름입니다.
- 서비스:
 - Direct Connect

product/disableactivationconfirmationemail

- 설명: 서비스 활성화를 확인하는 이메일을 보내는 기능을 활성화하거나 비활성화합니다.

product/durability

- 설명: 지정한 연도 동안의 객체 내구성을 설명합니다.
- 샘플 값: 99.999999999%, N/A, 99.99%
- 서비스:
 - Amazon Glacier
 - Amazon S3
 - AWS Elemental MediaStore

E

product/ebsOptimized

- 설명: Amazon EC2 인스턴스가 Amazon EBS에 최적화되어 있는지 여부를 설명합니다.
- 샘플 값: Yes, No
- 서비스:
 - Amazon EC2

product/ecu

- 설명: Amazon EC2 인스턴스의 정수 처리 파워에 대한 상대적인 척도를 제공하는 EC2 컴퓨팅 유닛 (ECU)에 대해 설명합니다.
- 샘플 값: 9, 100, variable
- 서비스:

- Amazon EC2
- OpenSearch Service
- Amazon GameLift Servers
- Amazon Redshift

product/endpointtype

- 설명: 디바이스가 연결되는 원격 연결의 특성을 설명합니다.

예: REST (표현적 상태 전송) 엔드포인트. REST API(또는 RESTful API)는 REST 아키텍처 스타일의 제약을 준수하는 애플리케이션 프로그래밍 인터페이스이며 RESTful 웹 서비스와 상호 작용할 수 있습니다.

- 샘플 값: Ipsec, Amazon SQS, AWS Lambda
- 서비스:
 - Amazon SNS
 - Amazon VPC
 - Storage Gateway
 - Amazon Glacier

product/enhancedNetworkingSupported

- 설명: 인스턴스가 향상된 네트워킹을 지원하는지 여부를 설명합니다. 향상된 네트워킹에서는 지원되는 인스턴스 유형에서 단일 루트 I/O 가상화(SR-IOV)를 사용하여 고성능 네트워킹 기능을 제공합니다.
- 샘플 값: Yes, No
- 서비스:
 - Amazon DocumentDB
 - Amazon EC2
 - Amazon Neptune
 - Amazon RDS
 - AWS 데이터베이스 마이그레이션 서비스

F

product/filesystemtype

- 설명: 로컬 또는 원격 스토리지 디바이스의 세부 정보와 운영 체제의 사양을 설명합니다.

product/findingGroup

- 설명: Security Hub에 저장된 조사 결과가 유료인지 무료인지를 지정합니다. 무료인 경우 이유도 명시할 수 있습니다.
- 샘플 값: FreeFindingsIngestion-CrossRegion, FreeFindingsIngestion-FreeTier, FreeFindingsIngestion-FreeTrial, PaidFindingsIngestion
- 서비스:
 - AWS Security Hub

product/findingSource

- 설명: 조사 결과가 Security Hub 컨트롤에서 생성되었는지 다른 파트너 보안 제품에서 생성되었는지 지정합니다.
- 샘플 값: SecurityHubProduct, OtherProduct
- 서비스:
 - AWS Security Hub

product/freeUsageIncluded

- 설명: AWS 프리 티어에 따른 무료 사용량은 매달 모든 리전의 무료 사용량을 합산하여 청구서에 자동으로 적용됩니다. 예를 들어, 사용하는 모든 리전에서 750시간의 Amazon EC2 Linux 마이크로 인스턴스를 무료로 받을 수 있습니다. 리전당 750시간이 아닙니다.
- 서비스:
 - Amazon Inspector –

product/fromLocation

- 설명: 사용이 시작된 위치를 설명합니다.
- 샘플 값: External, US East (N. Virginia), Global

- 서비스:
 - Amazon CloudFront
 - AWS DataTransfer

product/fromRegionCode

- 설명: AWS 서비스의 소스 리전 코드를 설명합니다. 자세한 내용은 [product/regioncode](#) 단원을 참조하십시오.
- 샘플 값: ap-northeast-1
- 서비스:
 - Amazon RDS
 - Amazon EC2
 - Amazon VPC
 - Direct Connect

product/fromLocationType

- 설명: 사용이 시작된 위치 유형을 설명합니다.
- 샘플 값: AWS Region, AWS Edge Location
- 서비스:
 - Direct Connect
 - AWS Elemental MediaConnect
 - Amazon CloudFront
 - Amazon Lightsail
 - AWS Shield

G

product/gpu

- 설명: GPU의 수를 설명합니다.
- 샘플 값: 16, 32
- 서비스:

- Amazon SageMaker AI
- Amazon EC2

product/gpuMemory

- 설명: GPU 메모리 세부 정보를 설명합니다.
- 샘플 값: 16, 32
- 서비스:
 - Amazon SageMaker AI
 - Amazon EC2

product/group

- 설명: 정의상 유사하거나 함께 그룹화된 여러 제품의 구조입니다. 예를 들어 Amazon EC2 팀은 제품을 공유 인스턴스, 전용 호스트, 전용 사용으로 분류할 수 있습니다.
- 서비스:
 - AWS Certificate Manager
 - AWS CodeCommit
 - AWS Glue
 - AWS IoT 분석
 - AWS Lambda

product/groupdescription

- 설명: 제품 그룹에 부여되는 간단한 이름입니다.
- 서비스:
 - AWS 예산
 - AWS Certificate Manager
 - AWS Lambda
 - Amazon SQS

|

product/insightsType

- 설명: 생성된 Insight 이벤트의 유형을 나타냅니다.
- 샘플 값: APICallVolume
- 서비스:
 - CloudTrail

product/instance

- 설명: Amazon EC2 인스턴스는 AWS 인프라에서 애플리케이션을 실행하기 위한 Amazon Elastic Compute Cloud(Amazon EC2)에 있는 가상 서버입니다. 에서 제공하는 AMI AWS, 사용자 커뮤니티 또는를 통해 선택할 수 있습니다 AWS Marketplace.
- 샘플 값: T3
- 서비스:
 - Amazon EC2

product/instanceFamily

- 설명: Amazon EC2 인스턴스 패밀리를 설명합니다. Amazon EC2는 10개의 서로 다른 인스턴스 유형에 대해 다양한 옵션을 제공합니다. 유형마다 하나 이상의 크기 옵션이 제공되며, 다양한 애플리케이션 유형에 맞게 최적화된 개별 인스턴스 패밀리로 구성되어 있습니다.
- 샘플 값: General Purpose, Memory Optimized, Accelerated Computing
- 서비스:
 - Amazon EC2
 - Amazon RDS
 - OpenSearch Service
 - Amazon ElastiCache
 - Amazon EMR

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/instanceSize

- 설명: 리소스의 인스턴스 크기를 나타냅니다.
- 샘플 값: 2vCPU, 4vCPU, 8vCPU, 16vCPU
- 서비스:
 - Amazon CodeCatalyst

product/instanceType

- 설명: 인스턴스의 CPU, 네트워킹 및 스토리지 용량을 정의하는 인스턴스 유형, 크기 및 패밀리를 설명합니다.
- 샘플 값: t2.small, m4.xlarge, t2.micro, m4.large, t2.large
- 서비스:
 - Amazon EC2
 - Amazon RDS
 - OpenSearch Service
 - Amazon ElastiCache
 - Amazon EMR

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/instanceTypeFamily

- 설명: 특정 사용량과 연결된 인스턴스 패밀리입니다.
- 샘플 값: t2, m4, m3
- 서비스:
 - Amazon DocumentDB
 - Amazon RDS

product/integratingApi

- 설명: Amazon API Gateway와 같은 서비스를 AWS 사용하는 애플리케이션 통합 또는 Amazon AppFlow를 사용하는 코드 없는 통합.

product/integratingService

- 설명:의 애플리케이션 통합 AWS 은 마이크로 서비스, 분산 시스템 및 서버리스 애플리케이션 내에 서 분리된 구성 요소 간에 통신하는 데 사용되는 서비스 제품군입니다. 전체 아키텍처를 리팩터링할 필요는 없습니다. 규모에 상관없이 애플리케이션을 분리하면 변경의 영향을 줄여 더 쉽게 업데이트 하고 새 기능을 더 빠르게 출시할 수 있습니다.

product/intelAvxAvailable

- 설명: 프로세스에 Intel Advanced Vector Extension 명령 세트가 있는지 여부를 설명합니다.
- 샘플 값: Yes, No
- 서비스:
 - Amazon EC2

product/intelAvx2Available

- 설명: 프로세스에 Intel Advanced Vector Extension 명령 세트가 2개 있는지 여부를 설명합니다.
- 샘플 값: Yes, No
- 서비스:
 - Amazon EC2

product/intelTurboAvailable

- 설명: 코어가 Intel Turbo Technology를 사용하여 주파수를 높이도록 허용하는지 여부를 설명합니다.
- 샘플 값: Yes, No
- 서비스:
 - Amazon EC2

product/invocation

- 설명: EventBridge 스케줄러가 API 또는 서비스에 수행하는 간접 호출에 대해 설명합니다.
- 샘플 값: Scheduled Invocation
- 서비스:
 - Amazon CloudWatch Events

L

product/licenseModel

- 설명: 인스턴스의 라이선스 모델을 설명합니다.
- 샘플 값: license-included, bring-your-own-license, general-public-license
- 서비스:
 - Amazon 앱Stream
 - Amazon EC2
 - Amazon MQ
 - Amazon Neptune
 - Amazon RDS

product/location

- 설명: Amazon S3 버킷이 있는 리전을 설명합니다.
- 샘플 값: Asia Pacific (Mumbai), Asia Pacific (Seoul), Canada (Central), EU (London), US West (Oregon)
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/locationType

- 설명: 작업의 엔드포인트를 설명합니다.
- 샘플 값: AWS Region, AWS Edge Location, Other
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager

- Amazon S3
- Amazon RDS
- Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/logsDestination

- 설명: AWS::Logs::Destination 리소스는 CloudWatch Logs 대상을 지정합니다. 대상은 물리적 리소스(예: Amazon Kinesis Data Streams)를 포함하고 해당 리소스를 로그 이벤트 스트림에 구독할 수 있습니다.
- 샘플 값: AWS Region, AWS Edge Location, Other
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

M

product/maxIopsBurstPerformance

- 설명: Amazon EBS 볼륨의 최대 IOPS 버스트 성능을 설명합니다.
- 샘플 값: 3000 IOPS for volumes <= 1TB
- 서비스:
 - Amazon EC2

product/maxIopsVolume

- 설명: Amazon EBS 볼륨의 초당 최대 입력/출력을 설명합니다.
- 샘플 값: 16,000 (maxIops for a General Purpose SSD (gp2))

- 서비스:
 - Amazon EC2

product/maxThroughputvolume

- 설명: Amazon EBS 볼륨의 최대 네트워크 처리량 볼륨을 설명합니다.
- 샘플 값: 500 MiB/s, 250 MiB/s, 1000 MiB/s, 40 - 90 MB/sec
- 서비스:
 - Amazon EC2
 - Amazon SageMaker AI

product/memory

- 설명: 컴퓨터가 신속하게 응답하는 데 필요한 지침 및 데이터를 위한 플레이스홀더 전자 장치입니다. 컴퓨터 바이트는 스토리지 단위를 나타냅니다.
- 서비스:
 - AWS Database Migration Service
 - DynamoDB Accelerator
 - Amazon DocumentDB
 - Amazon EC2

product/messageCountfee

- 설명: 사용량이 메시지 수 또는 부과된 요금을 나타내는 지 여부를 나타내는 미터링 사용 유형을 설명합니다.
- 샘플 값: CarrierFeeCount, MessageFees, MessageCount, CarrierFees
- 서비스:
 - Amazon Simple Notification Service

product/messageType

- 설명: SMS 메시지의 유형을 설명합니다. SNS는 아웃바운드 SMS만 지원한다는 점에 유의하세요.
- 샘플 값: OutboundSMS

- 서비스:
 - Amazon Simple Notification Service

N

product/networkPerformance

- 설명: Amazon EC2 인스턴스의 네트워크 처리량을 설명합니다.
- 샘플 값: moderate, high, up to 10 GB
- 서비스:
 - Amazon EC2
 - Amazon RDS
 - Amazon ElastiCache
 - Amazon SageMaker AI
 - AWS 데이터베이스 마이그레이션 서비스

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/normalizationSizeFactor

- 설명: 인스턴스 크기의 정규화 인자를 설명합니다.
- 샘플 값: nano - 0.25, micro - 0.5, medium - 2, xlarge - 8, 16xlarge - 128
- 서비스:
 - Amazon DocumentDB
 - Amazon EC2
 - Amazon MQ
 - Amazon Neptune
 - Amazon RDS

O

product/operatingSystem

- 설명: Amazon EC2 인스턴스의 운영 체제를 설명합니다.

- 샘플 값: Amazon Linux, Ubuntu, Windows Server, Oracle Linux, FreeBSD
- 서비스:
 - Amazon 앱Stream
 - Amazon EC2
 - Amazon GameLift Servers
 - Amazon Lightsail
 - Amazon WorkSpaces
 - AWS CodeBuild

product/operation

- 설명: 이 항목에서 다루는 특정 AWS 작업을 설명합니다.
- 샘플 값: RunInstances(Amazon EC2 인스턴스 작업을 나타냄)
- 서비스:
 - Amazon EC2
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB
 - Amazon CloudWatch
 - Amazon Redshift

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/originIdType

- 설명: SMS 메시지를 보낼 때 사용되는 원본 ID의 유형을 설명합니다.
- 샘플 값: Sharedroute
- 서비스:
 - Amazon Simple Notification Service

product/osType

제품 설명 정보 리소스의 운영 체제를 설명합니다.

- 샘플 값: Dev Environment, Linux, Linux ARM64, Windows
- 서비스:
 - Amazon CodeCatalyst

P

product/parameterType

- 설명: 스택을 생성하거나 업데이트 할 때 CloudFormation 에서 파라미터를 사용하여 템플릿에 사용자 지정 값을 입력합니다. 예: InstanceTypeParameter. 스택을 생성 또는 업데이트할 때 이 파라미터를 사용하여 Amazon EC2 인스턴스 유형을 지정할 수 있습니다.

product/physicalCores

- 설명: 인스턴스가 제공하는 물리적 코어 수를 설명합니다.
- 샘플 값: 4, 8
- 서비스:
 - Amazon EC2

product/physicalProcessor

- 설명: Amazon EC2 인스턴스의 프로세서를 설명합니다.
- 샘플 값: High Frequency Intel Xeon E7-8880 v3 (Haswell), Intel Xeon E5-2670, AMD EPYC 7571
- 서비스:
 - Amazon DocumentDB
 - Amazon EC2
 - Amazon Neptune
 - Amazon RDS
 - AWS 데이터베이스 마이그레이션 서비스

product/platoClassificationType

- 설명: 데이터 주석 워크플로 라우팅에 대한 객체별 계층별 요금입니다.

- 샘플 값: LabeledObject, 3DLabeledObjectMultiFrame, 3DLabeledObject, Processing:VolumeUsage
- 서비스:
 - Amazon SageMaker AI

product/pricingUnit

- 설명: AWS 서비스의 가장 작은 결제 단위입니다. 예를 들어 API 직접 호출당 0.01c입니다.
- 서비스:
 - Directory Service

product/primaryplaceofuse

- 설명: 고객의 서비스 사용이 주로 발생하는 주요 비즈니스 또는 거주지 주소 위치입니다.

product/processorArchitecture

- 설명: 프로세서 아키텍처를 설명합니다.
- 샘플 값: 32-bit, 64-bit
- 서비스:
 - Amazon DocumentDB
 - Amazon EC2
 - Amazon Neptune
 - Amazon RDS
 - AWS 데이터베이스 마이그레이션 서비스

product/processorFeatures

- 설명: 인스턴스의 프로세서 기능을 설명합니다.
- 샘플 값: Intel AVX, Intel AVX2, Intel AVX512, Intel Turbo
- 서비스:
 - AWS 데이터베이스 마이그레이션 서비스
 - Amazon DocumentDB

- Amazon EC2
- Amazon Neptune
- Amazon RDS

product/ProductFamily

- 설명: 제품 유형의 범주입니다.
- 샘플 값: Alarm, AWS Budgets, Stopped Instance, Storage Snapshot, Compute
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/ProductName

- 설명: AWS 서비스의 전체 이름입니다. 이 열을 사용하여 AWS 서비스별로 AWS 사용량을 필터링합니다.
- 샘플 값: AWS Backup, AWS Config, Amazon Registrar, Amazon Elastic File System, Amazon Elastic Compute Cloud

product/productSchemaDescription

- 설명: 제품 구성 방식에 대한 청사진입니다. 여기에는 제품을 구성하는 다양한 속성이 포함됩니다.

product/provisioned

- 설명: Amazon EBS 사용량이 프로비저닝된 Amazon EBS 스토리지와 관련되어 있는지 여부를 나타냅니다.
- 샘플 값: Yes, No
- 서비스:

- Amazon EC2
- Amazon MQ

product/provisioningType

- 설명: 리소스가 온디맨드로 배포되었는지 또는 사전 프로비저닝되었는지 설명합니다.
- 샘플 값: On-Demand, Pre-Provisioned
- 서비스:
 - Amazon CodeCatalyst

product/PurchaseOption

- 설명: AWS 서비스에 사용할 수 있는 구매 모델을 설명합니다. 예를 들어,는의 추가 옵션과 함께 On-Demand, Spot Instances, Reserved Instances의 네 가지 기본 Amazon EC2 인스턴스 구매 옵션을 AWS 제공합니다Dedicated Hosts.

product/purchaseterm

- 설명: Amazon EC2에서는 이를 통해 일관된 인스턴스 구성을 약정할 것을 지정합니다. 여기에는 1~3년 기간의 인스턴스 유형 및 리전이 포함됩니다.

R

product/region

- 설명: AWS 서비스를 호스팅하는 지리적 영역입니다. 이 필드를 사용하여 특정 리전에 걸쳐 사용량을 분석합니다.
- 샘플 값: eu-west-3, us-west-1, us-east-1, ap-northeast-2, sa-east-1
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/regioncode

- 설명: 리전은 데이터 센터가 클러스터링되는 전 세계의 물리적 위치입니다.는 논리적 데이터 센터의 각 그룹을 가용 영역(AZ)으로 AWS 호출합니다. 각 AWS 지역은 지리적 영역 내에 물리적으로 분리된 여러 개의 격리된 AZ로 구성됩니다. 리전 코드 속성은 AWS 리전과 이름이 동일하며 AWS 서비스를 사용할 수 있는 위치를 지정합니다.
- 샘플 값: us-west-2, us-east-1, ap-southeast-2
- 서비스:
 - Amazon SageMaker AI

product/replicationType

- 설명: 서비스를 무료로 사용할 수 있도록 지정합니다. 예를 들어 AWS Server Migration Service 는 무료로 사용할 수 있으며 마이그레이션 프로세스 중에 사용된 스토리지 리소스에 대해서만 비용을 지불하면 됩니다.
- 샘플 값: Free
- 서비스:
 - AWS 애플리케이션 마이그레이션 서비스

product/resourceAssessment

- 설명: 증거를 수집, 저장 및 관리하는 프로세스입니다. 이를 사용하여 산업 표준 및 규정의 위험과 준수 여부를 평가할 수 있습니다.
- 샘플 값: All assessment
- 서비스:
 - AWS Audit Manager

product/resourcePriceGroup

- 설명: 리소스 유형, 리소스 및 가격 그룹(모니터링에 부과되는 가격, 현재 A와 B의 두 가지 가격 등급이 있음)을 설명합니다. 따라서 예를 들어 RDS 리소스를 모니터링할 경우 리소스 유형은 RDS('제품'), 리소스는 인스턴스, 가격 그룹은 B가 됩니다.

- 샘플 값: RDS-DBInstance-GroupB
- 서비스:
 - Amazon DevOps Guru

product/routeType

- 설명: 사용된 SMS 경로 유형을 설명합니다. 현재는 표준만 적용됩니다.
- 샘플 값: Standard
- 서비스:
 - Amazon Simple Notification Service

S

product/servicecode

- 설명: 고객에 대한 특정 AWS 서비스를 고유한 짧은 약어로 식별합니다.
- 샘플 값: Amazon EC2, AWS KMS
- 서비스:
 - AWS 예산
 - AWS Backup
 - AWS Certificate Manager
 - AWS Cloud Map
 - AWS CloudTrail

product/servicename

- 설명: AWS 서비스에 대한 간단한 설명입니다.
- 서비스:
 - Amazon EC2 예산
 - Amazon ECR
 - Amazon ECS
 - Amazon EFS
 - Amazon Elastic Inference

- Amazon EKS

product/singleOrDualPass

- 설명: 동영상에 발생하는 인코딩 유형을 결정하는 데 사용되는 용어입니다. 1패스 인코딩에서는 첫 번째 패스 자체에서 인코딩이 수행됩니다. 2패스 인코딩의 경우 첫 번째 패스에서 파일을 철저하게 분석하여 중간 파일을 생성합니다. 두 번째 패스에서 인코더는 중간 파일을 찾아 비트를 할당합니다. 실제 인코딩은 두 번째 패스에서 수행됩니다.
- 서비스:
 - AWS Elemental MediaConvert

product/sizeFlex

- 설명: RI의 정규화된 이점을 리전 및 인스턴스 패밀리 내의 다른 인스턴스 크기에도 적용할 수 있는지 여부를 설명합니다.
- 샘플 값: true, false
- 서비스:
 - - Amazon Elastic Compute Cloud

product/sku

- 설명: 제품의 고유 코드입니다. SKU는 ProductCode, UsageType 및 Operation을 결합하여 생성됩니다. 유연한 크기의 RI의 경우 SKU가 사용되었던 인스턴스를 사용합니다. 예를 들어 t2.micro 인스턴스를 사용하고 사용량에 t2.small RI 할인을 AWS 적용한 경우를 사용하여 품목 SKU가 생성됩니다t2.micro.
- 샘플 값: FFNT87MQSCR328W6, VBYCEU494XUAHCA7
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/storage

- 설명: 인스턴스에 연결된 디스크 스토리지를 설명합니다.
- 샘플 값: 60GB, True, EBS Only, 1 x 900 NVMe SSD, 1 x 150 NVMe SSD
- 서비스:
 - Amazon EC2
 - Amazon RDS
 - Amazon Redshift
 - OpenSearch Service
 - Amazon WorkSpaces

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/storageclass

- 설명: Amazon S3 버킷의 스토리지 클래스를 설명합니다.
- 샘플 값: Archive, General Purpose, Infrequent Access, Intelligent-Tiering, Non-Critical Data
- 서비스:
 - AWS Elemental MediaStore
 - AWS Storage Gateway
 - Amazon Cloud Directory
 - Amazon EFS
 - Amazon MQ
 - Amazon S3

product/storagemedia

- 설명: 저장 매체는 전자 데이터를 배치, 보관 및 검색하는 데 사용되는 디바이스 및 재료를 포함한 모든 기술입니다.
- 서비스:
 - AWS Database Migration Service
 - Amazon CloudWatch

- Amazon DocumentDB
- Amazon EC2
- Amazon ES

product/storagetype

- 설명: 컴퓨터가 정보를 저장하는 방법과 위치를 설명합니다. 컴퓨터, 서버 또는 컴퓨팅 디바이스의 내부 또는 외부일 수 있습니다.
- 샘플 값: Amazon S3, SSD, SSD-backed
- 서비스:
 - AWS Backup
 - Amazon ECR

T

product/tenancy

- 설명: Amazon EC2 인스턴스에 허용되는 테넌시 유형입니다.
- 샘플 값: Dedicated, Reserved, Shared, NA, Host
- 서비스:
 - Amazon EC2
 - Amazon ECS

product/throughputCapacity

- 설명: 파일 시스템을 호스팅하는 파일 서버가 파일 데이터를 제공할 수 있는 속도를 설명합니다. Amazon FileCache의 경우 값은 1000만 됩니다.
- 샘플 값: 12, 40, 50, 100, 125, 250, 500, 1000
- 서비스:
 - Amazon FileCache
 - Amazon FSx

product/tier

- 설명:를 AWS사용하면 사용량이 증가함에 따라 볼륨 기반 할인 및 절감 효과를 얻을 수 있습니다. Amazon S3와 같은 서비스의 경우 요금이 계층별로 책정됩니다. 즉, 더 많이 사용할수록 GB당 지불하는 금액이 줄어듭니다.는 비즈니스 요구 사항을 지원하는 서비스를 획득할 수 있는 옵션을 AWS 제공합니다.
- 서비스:
 - AWS Elemental MediaConvert

product/toLocation

- 설명: 위치 사용 대상을 설명합니다.
- 샘플 값: External, US East (N. Virginia)
- 서비스:
 - Amazon CloudFront
 - AWS 데이터 전송

product/toLocationType

- 설명: 서비스 사용의 대상 위치를 설명합니다.
- 샘플 값: AWS Region, AWS Edge Location
- 서비스:
 - Direct Connect
 - AWS Elemental MediaConnect
 - AWS Shield
 - Amazon CloudFront
 - Amazon Lightsail

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/toRegionCode

- 설명: AWS 서비스의 소스 리전 코드를 설명합니다. 자세한 내용은 [product/regioncode](#) 단원을 참조하십시오.

- 샘플 값: eu-west-1
- 서비스:
 - Amazon RDS
 - Amazon EC2
 - Amazon VPC
 - Direct Connect

product/transcodingResult

- 설명: 인코딩된 동영상 소스를 압축되지 않은 중간 형식으로 디코딩하고 대상 형식으로 다시 인코딩한 결과입니다.
- 서비스:
 - AWS Elemental MediaConvert

product/trialProduct

- 설명: AWS CloudHSM 에 여유 시간이 허용되는지 설명합니다.
- 서비스:
 - AWS CloudHSM

U

product/upfrontCommitment

- 설명: 사용량 약정이 필요한지 여부를 설명합니다 AWS CloudHSM. HSM이 AWS CloudHSM 클러스터에 프로비저닝되는 각 시간 (또는 일부 시간)에 대해 시간당 요금이 부과됩니다. HSM이 없는 클러스터에는 요금이 청구되지 않으며 암호화된 백업의 자동 저장 비용도 청구되지 않습니다. 자세한 내용은 [AWS CloudHSM 요금](#)을 참조하세요.

HSM과 주고 받는 네트워크 데이터 전송에는 별도의 요금이 부과됩니다. 자세한 내용은 [Amazon EC2 요금](#)을 참조하세요.

- 서비스:
 - AWS CloudHSM

product/usagetype

- 설명: 항목의 사용량 정보를 설명합니다.
- 샘플 값: EU-BoxUsage:c5d.9xlarge, EU-BoxUsage:m4.16xlarge, SAE1-InstanceUsage:db.t2.medium, USW2-AW-SW-19, SAE1-BoxUsage:c4.large
- 서비스:
 - Amazon EC2
 - AWS Certificate Manager
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

V

product/vcpu

- 설명: 단일 CPU 코어에서 동시에 실행되는 스레드 수를 설명합니다. Amazon EC2 인스턴스는 여러 개의 스레드를 하나의 CPU 코어에서 동시에 실행할 수 있도록 하는 멀티스레딩을 지원합니다. 각 스레드는 인스턴스에서 가상 CPU(vCPU)로 표현됩니다.
- 샘플 값: 8, 16, 36, 72, 128
- 서비스:
 - Amazon EC2
 - Amazon RDS
 - Amazon Redshift
 - OpenSearch Service
 - Amazon ElastiCache

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

product/videoCodec

- 설명: 디지털 동영상을 압축 및 압축 해제하는 소프트웨어 또는 하드웨어입니다. 비디오 압축의 관점에서 코덱은 인코더와 디코더를 혼합한 것입니다. 압축만 하는 디바이스를 일반적으로 인코더라고 하고 압축만 푸는 디바이스를 디코더라고 합니다.
- 서비스:
 - AWS Elemental MediaConvert;

product/videoFrameRate

- 설명: 동영상 프레임 속도(초당 프레임(FPS)으로 표시됨)는 동영상 카메라, 컴퓨터 그래픽 및 모션 캡처 시스템에서 연속적인 이미지(프레임)를 캡처하거나 표시하는 주파수 속도입니다.
- 서비스:
 - AWS Elemental MediaConvert;

product/videoQualitySetting

- 설명: 인코딩에 사용되는 품질 설정에 대해 설명합니다. 이 설정은 압축 효율에 영향을 미치고, 따라서 주어진 비트레이트에서의 비디오 품질에도 영향을 미칩니다.
- 샘플 값: Multi-pass, Multi-pass HQ, NA, Single-pass, Single-pass HQ
- 서비스:
 - AWS Elemental MediaConvert

product/volumeType

- 설명: Amazon EBS 볼륨 유형을 설명합니다.
- 샘플 값: Standard, General Purpose, General Purpose-Aurora, Amazon Glacier, Amazon SimpleDB - Standard
- 서비스:
 - Amazon EC2
 - Amazon S3
 - Amazon RDS
 - Amazon DynamoDB
 - Amazon Glacier

등. 전체 서비스 목록을 보려면 [Column_Attribute_Service.zip](#)을 다운로드하십시오.

W

product/workforceType

- 설명: 고용 또는 실업자 인력풀의 세분화입니다. 예: Full Time Employees(FTE) 또는 Temporary.
- 서비스:
 - Amazon SageMaker AI

리소스 태그 세부 정보

AWS 비용 및 사용 보고서의 resource 열을 사용하여 행 항목이 다루는 특정 리소스에 대한 정보를 찾을 수 있습니다. 이 칼럼은 사용자 정의 비용 할당 태그를 포함합니다. 예는 다음과 같습니다.

resourceTags/user:creator

user:Creator 태그를 사용하여 사용자가 리소스를 생성했는지 추적합니다.

resourceTags/user:name

user:Name 태그를 사용하여 특정 사용자가 어떤 리소스와 연관되어 있는지 추적합니다.

resourceTags/user:owner

user:Owner 태그를 사용하여 어느 사용자가 리소스를 가지고 있는지 추적합니다.

resourceTags/user:purpose

user:Purpose 태그를 사용하여 리소스가 왜 생성되었는지 추적합니다.

절감형 플랜 세부 정보

AWS 비용 및 사용 보고서의 SavingsPlan 열은 Savings Plans에 대한 세부 정보를 제공합니다. 절감형 플랜에 대한 자세한 정보는 절감형 플랜 사용 설명서의 [절감형 플랜이란?](#)을 참조하세요.

[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

- 설명: 결제 기간 동안 부과되는 절감형 플랜 구독에 대한 선결제 금액입니다. 이번 달 동안 분할 상환되는 전체 선결제 절감형 플랜 및 부분 선결제 절감형 플랜의 초기 선결제 금액입니다. 선결제 없음 절감형 플랜의 경우 값은 0입니다.
- 해당하는 항목: SavingsPlanRecurringFee
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

E

savingsPlan/EndTime

- 설명: 절감형 플랜 계약의 만료 날짜입니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

I

savingsPlan/InstanceTypeFamily

- 설명: 특정 사용량과 연결된 인스턴스 패밀리입니다.
- 적용 가능한 항목: SavingsPlanCoveredUsage
- 샘플 값: m4, g2

- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

N

savingsPlan/NetAmortizedUpfrontCommitmentForBillingPeriod

결제 기간 동안 부과되는 절감형 플랜 구독 선결제 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

savingsPlan/NetRecurringCommitmentForBillingPeriod

절감형 플랜 요금의 혼합되지 않은 순 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

savingsPlan/NetSavingsPlanEffectiveCost

절감형 플랜의 실질 비용은 사용량을 요금으로 나눈 값입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.

O

savingsPlan/OfferingType

- 설명: 구매한 절감형 플랜의 유형을 설명합니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 샘플 값: ComputeSavingsPlans, EC2InstanceSavingsPlans, SageMakerSavingsPlans
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

P

savingsPlan/PaymentOption

- 설명: 절감형 플랜에 사용 가능한 결제 옵션입니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 샘플 값: Partial Upfront, All Upfront, No Upfront
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

savingsPlan/PurchaseTerm

- 설명: 절감형 플랜의 기간 또는 약정을 설명합니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 샘플 값: 1yr, 3yr
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

R

savingsPlan/RecurringCommitmentForBillingPeriod

- 설명: 절감형 플랜 구독에 대한 월간 기본 요금입니다. 예를 들어 부분 선결제 절감형 플랜 또는 선결제 없음 절감형 플랜에 대한 월간 기본 요금입니다.
- 해당하는 항목: SavingsPlanRecurringFee
- 서비스:

- Amazon EC2
- Fargate
- AWS Lambda
- Amazon SageMaker AI

savingsPlan/Region

- 설명: AWS 서비스를 호스팅하는 AWS 리전(지리적 영역)입니다. 이 필드를 사용하여 특정 AWS 리전의 지출을 분석할 수 있습니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 샘플 값: US East (N. Virginia), US West (N. California), US East (Ohio), Asia Pacific (Mumbai), Europe (Ireland)
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

S

savingsPlan/SavingsPlanArn

- 설명: 고유한 절감형 플랜 식별자입니다.
- 해당하는 항목: SavingsPlanUpfrontFee
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

savingsPlan/SavingsPlanEffectiveCost

- 설명: 절감형 플랜 월간 약정 금액(선결제 및 기본 요금)의 비율은 각 사용량 행에 할당됩니다.
- 적용 가능한 항목: SavingsPlanCoveredUsage
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

savingsPlan/SavingsPlanRate

- 설명: 사용량에 대한 절감형 플랜 요금입니다.
- 적용 가능한 항목: SavingsPlanCoveredUsage
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

savingsPlan/StartTime

- 설명: 절감형 플랜 계약의 시작 날짜입니다.
- 해당하는 항목: SavingsPlanCoveredUsage, SavingsPlanNegation, SavingsPlanUpfrontFee, SavingsPlanRecurringFee
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

T

savingsPlan/TotalCommitmentToDate

- 설명: 해당 시간에 대한 지금까지의 총 분할 상환 선결제 약정 및 반복 청구 요금 약정입니다.
- 해당하는 항목: SavingsPlanRecurringFee
- 서비스:
 - Amazon EC2
 - Fargate
 - AWS Lambda
 - Amazon SageMaker AI

U

savingsPlan/UsedCommitment

- 설명: 사용된 절감형 플랜 약정의 총 금액(달러)입니다. (SavingsPlanRated와 사용량을 곱함)
- 해당하는 항목: SavingsPlanRecurringFee
- 서비스:
 - Amazon EC2
 - AWS Lambda
 - Fargate
 - Amazon SageMaker AI

비용 범주 세부 정보

AWS 비용 및 사용 보고서의 Cost Category 열을 사용하여 Cost Categories 기능으로 분류된 특정 행 항목에 대한 정보를 찾을 수 있습니다. Cost Category 열은 비용 범주 및 범주화 규칙을 생성할 때 자동으로 채워집니다. 이러한 열에는 열 헤더로 사용자 정의 비용 범주 이름과 함께 해당 비용 범주 값이 포함됩니다. 예는 다음과 같습니다.

costCategory/Project

프로젝트 비용 범주를 사용하여 프로젝트와 관련된 비용 및 사용 정보를 분류합니다. 'alpha' 또는 'beta'와 같은 프로젝트 이름은 costCategory/Project 아래에 값으로 표시됩니다.

costCategory/Team

팀 비용 범주를 사용하여 팀과 관련된 비용 및 사용 정보를 분류합니다. '마케팅'과 같은 팀 이름은 costCategory/Team 아래에 값으로 표시됩니다.

costCategory/Environment

환경 비용 범주를 사용하여 환경과 관련된 비용 및 사용 정보를 분류합니다. '개발'과 같은 환경 이름은 costCategory/Environment 아래에 값으로 표시됩니다.

할인 세부 정보

discount 헤더 아래의 열은 보고서의 결제 기간 동안 계정에 할인이 적용된 경우에만 AWS Cost & Usage Report(AWS CUR)에 포함됩니다. 여기에는 다음 열이 포함됩니다.

discount/BundledDiscount

번들 할인은 항목에 적용됩니다. 번들 할인은 다른 서비스 또는 기능의 사용에 따라 서비스 또는 기능을 무료 또는 할인된 가격으로 제공하는 사용량 기반 할인입니다. 번들 할인의 예는 다음과 같습니다.

- 를 사용하는 경우 별도로 비용을 지불할 AWS WAF 필요가 AWS Shield Advanced 없습니다. AWS WAF 사용량은 번들로 제공됩니다 AWS Shield Advanced. 에 대한 자세한 내용은 [Amazon CloudFront 요금](#)을 AWS Shield Advanced 참조하세요.
- 를 사용하여 NAT 게이트웨이를 생성하는 경우 AWS Network Firewall 방화벽의 GB 및 사용 시간당 처리와 함께 표준 NAT 게이트웨이 처리 및 시간당 사용 요금이 one-to-one로 면제됩니다. 자세한 내용은 [AWS Network Firewall 요금](#)을 참조하십시오.
- Amazon Interactive Video Service(IVS) 채팅을 사용하면 1시간의 비디오 입력을 전송할 때마다 추가 비용 없이 2,700개의 전송 메시지와 270,000개의 전달된 메시지를 받을 수 있습니다. 자세한 내용은 [Amazon Interactive Video Service 요금](#)을 참조하세요.

discount/TotalDiscount

해당 항목에 대한 모든 할인 열의 합계입니다.

분할 항목 세부 정보

AWS 비용 및 사용 보고서의 splitLineItem 헤더 아래에 있는 열은 비용 할당 데이터 분할 기능을 오픈한 경우 비용 및 사용 보고서에 표시되는 필드입니다. 자세한 내용은 [분할 비용 할당 데이터 이해](#)를 참조하세요. 이 기능은 Amazon ECS(Fargate 포함) AWS Batch 및 Amazon EKS로만 제한됩니다.

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [J](#) | [K](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [Q](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#) | [Y](#) | [Z](#)

A

splitLineItem/ActualUsage

- 설명: Amazon ECS 작업 또는 Kubernetes 포드를 위해 지정된 기간 동안 발생한 vCPU 또는 메모리 사용량(lineItem/UsageType 기준)입니다.
- 해당하는 항목: Usage
- 샘플 값: 0.1, 0.5, 1.3
- 서비스: Amazon ECS, Fargate, Amazon EKS

Note

Fargate 비용은 vCPU 및 메모리 예약을 기반으로 계산되며 lineItem/UsageAmount 열에 반영됩니다. 분할 비용 할당 데이터는 지연 시간이 짧은 원격 측정 데이터를 관찰한 다음 이를 시간, 일, 월별로 집계하여 실제 Fargate vCPU 및 메모리 사용량을 계산합니다. 따라서 splitLineItem/ActualUsage는 lineItem/UsageAmount와 동일하지 않을 수 있습니다.

N

splitLineItem/NetSplitCost

- 설명: 모든 할인이 적용된 후의 Amazon ECS 작업 또는 Kubernetes 포드에 대한 실질 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75
- 서비스: Amazon ECS, Fargate, Amazon EKS

splitLineItem/NetUnusedCost

- 설명: 모든 할인이 적용된 후의 Amazon ECS 작업 또는 Kubernetes 포드에 대한 실질 미사용 비용입니다. 이 열은 해당 결제 기간에 계정에 할인이 적용되는 경우에만 보고서에 포함됩니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75

- 서비스: Amazon ECS, Fargate, Amazon EKS

Note

미사용 비용은 splitLineItem/SplitUsage를 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 비례하여 적용됩니다.

P

splitLineItem/ParentResourceId

- 설명: Amazon ECS 작업 또는 Amazon EKS 포드와 연결된 상위 EC2 인스턴스의 리소스 ID입니다(lineItem/ResourceId 열에서 참조). 상위 리소스 ID는 지정된 기간 동안의 ECS 작업 또는 Kubernetes 포드 워크로드가 상위 EC2 인스턴스에서 실행되었음을 의미합니다. 이는 EC2 시작 유형의 Amazon ECS 작업 또는 Kubernetes 포드에만 적용됩니다.
- 해당하는 항목: Usage
- 서비스: Amazon ECS, Amazon EKS

Note

splitLineItem/ParentResourceId는 리소스 IDs AWS 비용 및 사용 보고서에 포함된 경우에만 사용할 수 있습니다.

splitLineItem/PublicOnDemandSplitCost

- 설명: 퍼블릭 온디맨드 인스턴스 요금(pricing/publicOnDemandRate 열에서 참조)을 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리 비용(lineItem/UsageType 기준)입니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75
- 서비스: Amazon ECS, Fargate, Amazon EKS

splitLineItem/PublicOnDemandUnusedCost

- 설명: 퍼블릭 온디맨드 인스턴스 요금을 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리의 미사용 비용(lineItem/UsageType 기준)입니다. 미사용 비용은 지정된 기간 동안 사용되지 않은 EC2 인스턴스(splitLineItem/ParentResourceId 열에서 참조)의 리소스(CPU 또는 메모리)와 관련된 비용입니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75
- 서비스: Amazon ECS, Fargate, Amazon EKS

R

splitLineItem/ReservedUsage

- 설명: Amazon ECS 작업 또는 Kubernetes 포드를 위해 지정된 기간 동안 구성된 vCPU 또는 메모리 사용량(lineItem/UsageType 기준)입니다.
- 해당하는 항목: Usage
- 샘플 값: 1, 2, 4
- 서비스: Amazon ECS, Fargate, Amazon EKS

S

splitLineItem/SplitCost

- 설명: Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리 비용(lineItem/UsageType 기준)입니다. 여기에는 EC2 인스턴스(splitLineItem/parentResourceId 열에서 참조)에 예약 또는 절감형 플랜에 대한 선결제 요금 또는 부분 선결제 요금이 있는 경우 분할 상환 비용이 포함됩니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75
- 서비스: Amazon ECS, Fargate, Amazon EKS

splitLineItem/SplitUsage

- 설명: 지정된 기간 동안 Amazon ECS 작업 또는 Kubernetes 포드에 할당된 vCPU 또는 메모리 사용량(lineItem/UsageType 기준)입니다. 이는 splitLineItem/ReservedUsage 또는 splitLineItem/ActualUsage의 최대 사용량으로 정의됩니다.
- 해당하는 항목: Usage
- 샘플 값: 1, 1.3
- 서비스: Amazon ECS, Fargate, Amazon EKS

splitLineItem/SplitUsageRatio

- 설명: Amazon ECS 작업 또는 Kubernetes 포드에 할당된 vCPU 또는 메모리(lineItem/UsageType 기준)와 EC2 인스턴스에서 사용 가능한 전체 CPU 또는 메모리의 비율(splitLineItem/ParentResourceId 열에서 참조)입니다.
- 해당하는 항목: Usage
- 샘플 값: 0.25, 0.75
- 서비스: Amazon ECS, Fargate, Amazon EKS

Note

splitLineItem/SplitUsageRatio는 시간별 데이터를 시간 단위로 기본 설정하는 AWS Cost & Usage Report(AWS CUR)에만 사용할 수 있습니다.

U

splitLineItem/UnusedCost

- 설명: Amazon ECS 작업 또는 Kubernetes 포드에 일정 기간 동안 할당된 vCPU 또는 메모리의 미사용 비용(lineItem/UsageType 기준)입니다. 미사용 비용은 지정된 기간 동안 사용되지 않은 EC2 인스턴스(splitLineItem/ParentResourceId 열에서 참조)의 리소스(CPU 또는 메모리)와 관련된 비용입니다. 여기에는 EC2 인스턴스(splitLineItem/parentResourceId)에 예약 또는 절감형 플랜에 대한 선결제 요금 또는 부분 선결제 요금이 있는 경우 분할 상한 비용이 포함됩니다.
- 해당하는 항목: Usage
- 샘플 값: 1.35, 1.75

- 서비스: Amazon ECS, Fargate, Amazon EKS

Note

미사용 비용은 splitLineItem/SplitUsage를 기준으로 Amazon ECS 작업 또는 Kubernetes 포드에 비례하여 적용됩니다.

사용 사례

AWS CUR(AWS 비용 및 사용 보고서)을 사용하여 보고서 관리 요구 사항에 맞게 조정할 수 있습니다. 이 섹션은 절감형 플랜 및 예약형 인스턴스(RI) 사용률, 요금 및 할당을 추적하는 등의 사용 사례를 이해하는 데 도움이 됩니다.

주제

- [절감형 플랜 이해](#)
- [예약 파악](#)
- [데이터 전송 요금 이해](#)
- [분할 비용 할당 데이터에 대한 이해](#)

절감형 플랜 이해

비용 및 사용 보고서(AWS CUR)를 사용하여 Savings Plans 사용률, 요금 및 할당을 추적할 수 있습니다.

절감형 플랜 항목

Savings Plans 1년 또는 3년 기간 동안 일관된 사용량(시간당 \$로 측정)을 약정하는 대가로 Amazon EC2 AWS Fargate AWS Lambda, 및 Amazon SageMaker AI에 대해 저렴한 가격을 제공하는 유연한 요금 모델을 제공합니다.

AWS CUR의 다음 항목은 Savings Plans.

Note

다음 표에서는 명확성을 위해 AWS CUR의 열과 행이 트랜스포지션됩니다. 첫 번째 열의 값은 보고서의 헤더를 나타냅니다. 이 예제에는 몇 개의 키 AWS CUR 열만 포함됩니다. 다른 AWS CUR 열에 대한 자세한 내용은 섹션을 참조하세요 [데이터 사전](#).

선납금

SavingsPlanUpfrontFee 항목은 All Upfront 또는 Partial Upfront 절감형 플랜을 구매하면 청구서에 추가됩니다. 다음 표는 이 일회성 요금이 일부 AWS CUR 열에 어떻게 표시되는지 보여줍니다.

lineItem/LineItemType	SavingsPlanUpfrontFee	SavingsPlanUpfrontFee
lineItem/ProductCode	ComputeSavingsPlans	ComputeSavingsPlans
lineItem/UsageStartDate	2019-10-10T00:03:54Z	2019-10-10T00:12:15Z
lineItem/LineItemDescription	1년 전체 선결제 컴퓨팅 절감형 플랜 ID: 70352035에 대해 43.80 USD 일회성 요금	3년 부분 선결제 모든 리전 M5 인스턴스 유형 EC2 절감형 플랜 ID: 12355516에 대해 43.80 USD 일회성 요금
lineItem/UnblendedCost	43.8	43.8
savingsPlan/SavingsPlanARN	arn:aws:savingsplans::555555555555:savingsplan/bc1d08fd	arn:aws:savingsplans::555555555555:savingsplan/67b0ef20

절감형 플랜 월 기본 사용료

SavingsPlanRecurringFee 항목은 No Upfront 또는 Partial Upfront 절감형 플랜에 해당하는 시간당 기본 요금을 설명합니다. SavingsPlanRecurringFee는 구매 당일에 청구서에 추가되고 이후에는 매 시간 추가됩니다.

시간(시간별 비용 및 사용량에 적용 가능) 또는 일(일별 비용 및 사용량에 적용 가능)에 할당된 SavingsPlanRecurringFee는 구매 시점에 청구서에 추가됩니다. 청구 기간의 각 시간/일이 나중에 추가됩니다.

All Upfront 절감형 플랜의 경우, 항목은 청구 기간 동안 사용하지 않은 절감형 플랜의 부분을 나타냅니다.

다음 표는 일부 AWS CUR 열에 반복 시간당 요금이 어떻게 표시되는지 보여줍니다.

lineItem/LineItemType	SavingsPlanRecurringFee	SavingsPlanRecurringFee
lineItem/UsageStartDate	2019-20-10T00:00:00Z	2019-20-10T00:00:00Z
lineItem/ProductCode	컴퓨팅 절감형 플랜	컴퓨팅 절감형 플랜
lineItem/UsageType	ComputeSP:1yrPartialUpfront	USE2-EC2SP:t3.1yrPartialUpfront
lineItem/UnblendedCost	0.01	0.01
lineItem/LineItemDescription	1년 부분 선결제 Compute Savings Plan	us-east-2에서 1년 부분 선결제 t3 EC2 Instance Savings Plan
savingsPlan/SavingsPlanARN	arn:aws:savingsplans::555555555555:savingsplan/bc1d08fd	arn:aws:savingsplans::555555555555:savingsplan/bc1d08fd
savingsPlan/PaymentOption	부분 선결제	부분 선결제
savingsPlan/OfferingType	ComputeSavingsPlans	EC2InstanceSavingsPlans
savingsPlan/PurchaseTerm	1년	1년

Note

SavingsPlanRecurringFee는 기본 RI 요금과 다르게 계산됩니다. 기본 RI 요금은 월별 요금이고 SavingsPlanRecurringFee는 시간당 요금입니다. 기본 RI 요금에 대한 자세한 정보는 [월 기본 RI 요금](#) 섹션을 참조하세요.

절감형 플랜 할인 혜택

SavingsPlanCoveredUsage 항목은 절감형 플랜 혜택을 받을 수 있는 인스턴스 사용량을 설명합니다. SavingsPlanCoveredUsage 항목은 절감형 플랜 혜택이 없었던 온디맨드 요금에 대한 일반 요금을 보여줍니다. 이 혼합되지 않은 요금은 해당 SavingsPlanNegation 항목에 의해 상쇄됩니다.

각 SavingsPlanCoveredUsage 항목에서 savingsPlan/SavingsPlanRate 및 savingsPlan/SavingsPlanEffectiveCost 필드를 사용하여 절감형 플랜의 시간당 약정을 기준으로 해당 사용량이 청구된 방식을 확인할 수 있습니다.

SavingsPlanCoveredUsage 항목에 해당하는 SavingsPlanNegation이 표시됩니다. SavingsPlanNegation 항목은 SavingsPlanCoveredUsage의 혼합되지 않은 비용을 상쇄하고 SavingsPlanARN, 작업, 사용 유형 및 가용 영역별로 시간별 수준에서 그룹화됩니다. 따라서 하나의 SavingsPlanNegation 항목은 여러 SavingsPlanCoveredUsage 항목에 해당될 수도 있습니다.

다음 표는 일부 AWS CUR 열에서 적용 사용량 및 부정 행 항목이 어떻게 나타나는지 보여줍니다.

lineItem/ LineItemType	SavingsPlanCovered Usage	SavingsPlanCovered Usage	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	온디맨드당 0.0052 USD - Linux t3.nano 인스턴스 시간	온디맨드당 0.0052 USD - Linux t3.nano 인스턴스 시간	AccountId별로 사용된 SavingsPlanNegotiation: 5555555555555555 및 UsageSkus : K7ERD2Q28 HHU97DT
SavingsPlan/ SavingsPlanARN	arn:aws:savingsplans:: 5555555555555555: savingsplan/bc1d08fd	arn:aws:savingsplans:: 5555555555555555: savingsplan/bc1d08fd	arn:aws:savingsplans:: 5555555555555555: savingsplan/bc1d08fd
savingsPlan/ SavingsPlanRate	0.0026	0.0026	
savingsPlan/ SavingsPlanEffectiveCost	0.0026	0.0013	

절감형 플랜 약정에서 총당할 수 있는 양보다 많은 사용량이 있는 경우에도 미적용 사용량은 여전히 사용량 항목으로 표시되고 보장 사용량은 해당 SavingsPlanNegotiation 항목과 함께 SavingsPlanCoveredUsage로 표시됩니다.

예약 파악

AWS CUR(AWS 비용 및 사용 보고서)을 사용하여 예약 인스턴스(RI) 사용률, 요금 및 할당을 추적할 수 있습니다. 이 섹션은 예약을 파악하기 위한 심층적인 설명입니다.

주제

- [예약 항목 이해](#)
- [분할 상환 예약 데이터 이해](#)
- [Amazon EC2에 대한 유연한 크기 예약 모니터링](#)
- [온디맨드 용량 예약 모니터링](#)

예약 항목 이해

RI는 온디맨드 인스턴스 요금과 비교하여 대폭 할인을 제공합니다. RI는 물리적인 인스턴스가 아닙니다. 계정에서 온디맨드 인스턴스를 사용할 때 적용되는 결제 할인입니다. 이러한 온디맨드 인스턴스가 결제 할인의 혜택을 받으려면 일정한 속성에 부합되어야 합니다.

주제

- [선납금](#)
- [트루업 요금](#)
- [월 기본 RI 요금](#)
- [RI 할인 혜택](#)
- [예약형 인스턴스 유형](#)
- [인스턴스 사용에 적용되는 예약형 인스턴스 혜택](#)

Note

다음 표에서는 명확성을 위해 AWS CUR의 열과 행이 트랜스포지션됩니다. 첫 번째 열의 값은 보고서의 헤더를 나타냅니다. 이 예제에는 몇 개의 키 AWS CUR 열만 포함됩니다. 다른 AWS CUR 열에 대한 자세한 내용은 섹션을 참조하세요 [데이터 사전](#).

선납금

All Upfront 또는 Partial Upfront RI를 구매하면 요금 항목이 청구서에 추가됩니다.

다음 표는 이 일회성 요금이 일부 AWS CUR 열에 어떻게 표시되는지 보여줍니다.

lineItem/LineItemType	Fee
lineItem/ProductCode	AmazonEC2
lineItem/UsageStartDate	2016-01-01T00:00:00Z
lineItem/LineItemDescription	구독 요금 등록: 363836886, planId: 1026576
lineItem/UnblendedCost	68

Reservation/ReservationARN	arn:aws:ec2:us-east-1:123456789012:reserved-instances/f8c204c1-dd48-43f1-adb8-f88aa61e0dea
----------------------------	--

트루업 요금

컨버터블 예약형 인스턴스를 교환하는 경우 기존의 예약형 인스턴스 및 새 예약형 인스턴스의 교환과 관련된 모든 비용(트루업 요금)도 요금 항목으로 청구서에 추가됩니다. 트루업 요금의 경우 reservation/ReservationARN 열에는 reserved-instance-exchange/riex가 포함됩니다.

다음 표에는 컨버터블 예약형 인스턴스 교환에 따른 트루업 요금이 나와 있습니다.

lineitem/ LineItemType	lineitem/ ProductCode	lineitem/ UsageStartDate	lineitem/ LineItemDescription	lineitem/ Unblended Cost	Reservation/ReservationARN
Fee	AmazonEC2	2016-01-01T00:00:00Z			arn:aws:ec2:eu-west-1:012345678901:reserved-instance-exchange/riex-example-5d71-4215-886f-17a3f64ea972

월 기본 RI 요금

[RI Fee] 항목에는 해당 월에 적용된 RI와 관련된 월 기본 요금이 표시됩니다. RI Fee(RI 요금)는 구매 당일에 청구서에 처음 추가되고 이후에는 매 결제 기간의 첫날에 추가됩니다.

[RI Fee]는 할인된 시간당 요금에 해당 월의 사용 시간을 곱한 값입니다.

다음 표를 통해 월 기본 요금이 보고서에 어떻게 표시되는지 알 수 있습니다.

lineitem/LineitemType	RI fee
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

가용 영역 또는 AWS 리전 리전 범위가 있는 RIs의 경우 반복 월별 요금이 다르게 기록됩니다. 가용 영역 범위가 있는 RI의 경우 해당 가용 영역이 reservation/AvailabilityZone 열에 표시됩니다. 리전 범위가 있는 RI의 경우 reservation/AvailabilityZone 열이 비어 있습니다. 리전 범위가 있는 RI에는 인스턴스 크기를 보여주는 lineitem/NormalizationFactor 및 reservation/TotalReservedNormalizedUnits 열의 값이 있습니다.

Note

기본 RI 요금은 SavingsPlanRecurringFee와 다르게 계산됩니다. 기본 RI 요금은 월별 요금이고 SavingsPlanRecurringFee는 시간당 요금입니다. SavingsPlanRecurringFee에 대한 자세한 정보는 [절감형 플랜 이해](#) 섹션을 참조하세요.

RI 할인 혜택

할인 사용량 행 항목은 일치하는 RI 할인 혜택을 받은 인스턴스 사용량을 설명하며, RIs 중 하나와 일치하는 사용량이 있는 경우 청구서에 추가됩니다. 는 인스턴스 예약과 일치하는 인스턴스 사용과 같은 일치하는 사용량을 기준으로 RI 할인 혜택을 AWS 계산합니다. 일치하는 사용이 있는 경우 사용량 항목과 관련된 비용은 항상 0입니다. 왜냐하면 RI와 관련된 요금은 이미 다른 두 항목(선결제 요금 및 월 기본 요금)에 표시되어 있기 때문입니다.

다음 표에서는 RI 할인 혜택을 받은 사용량의 예를 보여줍니다.

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 예약형 인스턴스 적용됨
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 DiscountedUsage 라인의 UsageAmount 값은 실제 사용 시간입니다.

[NormalizedUsageAmount]의 값은 [UsageAmount] 값에 [NormalizationFactor] 값을 곱한 것입니다. [NormalizationFactor]의 값은 인스턴스 크기에 따라 달라집니다. RI 혜택 할인이 사용량의 일치하는

항목에 적용되는 경우 초기 선결제 비용 및 월 기본 요금에 대한 reservation/ReservationARN 열의 Amazon 리소스 이름(ARN) 값은 할인 사용 항목의 ARN 값과 일치합니다.

인스턴스 크기를 정규화 인자에 매핑하는 방법에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 크기 수정을 위한 지원](#)을 참조하세요.

예약형 인스턴스 유형

보고서 항목이 표준 예약 인스턴스 또는 컨버터블 예약형 인스턴스와 연결되어 있는지 확인하려면 lineitem/LineItemType 열을 요금 또는 RI 요금으로 필터링합니다. 그런 다음 예약형 인스턴스 유형을 나타내는 product/OfferingClass 열을 검토합니다.

보고서 항목이 영역 또는 리전별 예약형 인스턴스와 연결되어 있는지 확인하려면 reservation/AvailabilityZone 열을 검토합니다. 영역 예약형 인스턴스의 경우 이 열에는 해당 가용 영역이 표시됩니다. 리전별 예약형 인스턴스의 경우 이 열은 비어 있습니다.

인스턴스 사용에 적용되는 예약형 인스턴스 혜택

다음 열 중 하나 이상을 기준으로 보고서를 필터링하여 어떤 인스턴스 사용 항목이 어떤 예약형 인스턴스의 혜택을 받았는지 파악할 수 있습니다.

- reservation/reservationARN: 이 열을 예약 ARN으로 필터링하여 각 항목과 관련된 예약형 인스턴스 임대를 식별합니다.
- lineitem/ResourceId: 이 열을 검토하여 예약형 인스턴스가 적용되는 리소스의 ID를 확인합니다.
- lineitem/LineItemType: 요금, RI 요금 또는 DiscountedUsage를 기준으로 이 열을 필터링하여 관련 요금 또는 혜택을 확인합니다.
- lineitem/UsageType: 이 열을 HeavyUsage 기준으로 필터링하여 RI 요금 항목을 식별합니다. 또는 이 열을 BoxUsage 기준으로 필터링하여 DiscountedUsage 항목을 식별합니다.
- lineitem/UsageAmount: RI 요금 항목의 경우 이 열에는 예약형 인스턴스가 적용된 달의 총 시간이 표시됩니다. DiscountedUsage 항목의 경우 이 열에는 보고서 구성 방식에 따라 예약형 인스턴스가 일별 또는 월별 수준에서 특정 인스턴스에 적용된 총 시간이 표시됩니다.

인스턴스 사용량에 적용되는 유연한 크기의 예약형 인스턴스의 정규화된 단위 수를 이해하려면 보고서의 lineitem/NormalizedUsageAmount 열을 검토합니다. 이 열의 값은 다음 열의 곱과 같습니다.

- lineitem/UsageAmount: 이 열에는 시간 단위로 측정된 인스턴스 사용량이 표시됩니다.
- lineitem/NormalizationFactor: DiscountedUsage 및 RI 요금 항목의 경우 이 열에는 인스턴스의 관련 정규화 인자가 표시됩니다. 정규화 인자에 대한 자세한 정보는 Amazon EC2 사용 설명서의 [정규화 인자에 의해 결정되는 인스턴스 크기 유연성](#)을 참조하세요.

계정이 여러 AWS Organizations 개인의 경우 예약 인스턴스에서 구매하거나 혜택을 받은 계정을 확인하려면 다음 열을 검토하세요.

- reservation/reservationARN: 예약 ARN을 검토하여 예약형 인스턴스를 구매한 계정을 확인합니다. ARN은 계정 ID를 포함합니다.
- lineitem/UsageAccountId: DiscountedUsage 항목의 경우 이 열은 구매한 예약형 인스턴스로부터 혜택을 받은 계정 ID를 식별합니다.

Note

예약형 인스턴스는 결제 구독이며 Amazon EC2 인스턴스와 같은 리소스가 아닙니다. 이 때문에 태그가 지정된 예약형 인스턴스는 태그가 지정된 리소스처럼 항목을 채우지 않습니다. DiscountedUsage가 있는 항목의 경우 태그는 예약형 인스턴스가 아닌 태그가 지정된 리소스에 채워집니다.

특정 예약형 인스턴스 임대와 관련된 비용을 식별하기 위해 예약형 인스턴스 ARN(임대 ID)을 기준으로 요금 또는 RI 요금 항목을 필터링할 수 있습니다. 예약 인스턴스에 대한 비용 데이터를 구성하려면 AWS Cost Categories를 사용하는 것이 좋습니다. 자세한 내용은 AWS Billing 사용 설명서의 [AWS Cost Categories를 사용한 비용 관리를 참조하세요.](#)

분할 상환 예약 데이터 이해

분할 상환은 일회성 예약 비용을 요금이 적용되는 결제 기간으로 분산시키는 것을 뜻합니다. 분할 상환하면 현금 기반이 아닌 요금 발생을 기반으로 요금을 계산해서 확인할 수 있습니다. 예를 들어 모든 선결제 RI에 대한 1년 요금이 365달러이고 해당 RI를 사용하는 인스턴스가 있는 경우 해당 인스턴스 요금은 하루에 1달러씩 분할 상환으로 결제됩니다.

청구 및 비용 관리가 분할 상환 비용을 계산하기 위해 사용하는 데이터를 다음의 비용 및 사용 보고서 열에서 확인할 수 있습니다.

주제

- [예약 인스턴스 인벤토리](#)
- [결제 기간에 대한 분할 상환 데이터](#)
- [예약 인스턴스 적용 요금](#)

Note

모든 예약형 인스턴스 항목에 모든 reservation/ 열이 채워지는 것은 아닙니다. 보고서의 reservation/ 열은 항목 유형을 기준으로 채워집니다. 예를 들어, RI 요금 항목이 reservation/UnusedAmortizedUpfrontFeeForBillingPeriod 열을 채웁니다. 한편, DiscountedUsage 항목이 reservation/effectivecost 열을 채웁니다.

예약 인스턴스 인벤토리

다음 열을 사용하여 RI 인벤토리를 추적할 수 있습니다. 이러한 열의 값은 RI 구독 항목(RI Fee 항목이라고도 함)에만 표시되며 RI를 사용하는 실제 인스턴스에는 표시되지 않습니다.

열 설명 및 샘플 값에 대한 자세한 내용은 [the section called “예약 세부 정보”](#) 단원을 참조하십시오.

- reservation/UpfrontValue
- reservation/startTime
- reservation/endTime
- reservation/modificationStatus

결제 기간에 대한 분할 상환 데이터

다음 열을 사용하여 결제 기간에 대한 RI의 분할 상환 요금을 확인할 수 있습니다. 이러한 열의 값은 RI 구독 항목(RI Fee 항목이라고도 함)에만 표시되며 RI를 사용하는 실제 인스턴스에는 표시되지 않습니다.

열 설명 및 샘플 값에 대한 자세한 내용은 [the section called “예약 세부 정보”](#) 단원을 참조하십시오.

- reservation/amortizedUpfrontFeeForBillingPeriod
- reservation/unusedQuantity
- reservation/unusedNormalizedUnitQuantity
- reservation/unusedRecurringFee
- reservation/unusedAmortizedUpfrontFeeForBillingPeriod

예약 인스턴스 적용 요금

다음 열을 사용하여 인스턴스 수준에서 적용 요금을 확인할 수 있습니다. 이러한 열의 값은 인스턴스 사용량 항목(Discounted Usage boxUsage 항목이라고도 함)에만 표시됩니다.

열 설명 및 샘플 값에 대한 자세한 내용은 [the section called “예약 세부 정보”](#) 단원을 참조하십시오.

- reservation/amortizedUpfrontCostForUsage
- reservation/recurringFeeForUsage
- reservation/effectiveCost

Amazon EC2에 대한 유연한 크기 예약 모니터링

한 리전에 적용되는 Amazon EC2 예약형 인스턴스에서는 가용 영역 유연성과 인스턴스 크기 유연성을 제공합니다. 가용 영역 유연성을 제공하는 예약형 인스턴스에서는 해당 리전에 속한 모든 가용 영역의 사용량에 대해 할인 혜택을 제공합니다. 인스턴스 크기 유연성을 제공하는 예약형 인스턴스는 해당 패 밀리 내의 인스턴스 크기와 상관없이 사용량에 대해 할인을 제공합니다. 크기가 유연한 예약형 인스턴스는 가장 작은 인스턴스 크기부터 적용됩니다. 자세한 내용은 Amazon EC2 사용 설명서의 [예약형 인스턴스 적용 방법](#)을 참조하세요.

예약형 인스턴스에서 제공하는 인스턴스 크기 유연성이 사용량에 어떻게 적용되는지 이해하려면 `lineltem/NormalizationFactor` 및 `lineltem/NormalizedUsageAmount` 열을 참조하세요.

Note

인스턴스 크기 유연성은 리전에 할당되는 기본 테넌시가 있는 Linux 또는 Unix 예약형 인스턴스에서만 지원됩니다. 리전별 예약형 인스턴스의 인스턴스 크기 유연성 제한에 대한 자세한 정보는 Amazon EC2 사용 설명서의 [리전별 예약형 인스턴스 적용 방법](#)을 참조하세요.

비용 및 사용 보고서에서는 예약형 인스턴스를 구매한 계정에 예약형 인스턴스 사용이 기본적으로 적용됩니다. 구매 계정에서 한 시간 내에 사용할 수 없는 예약형 인스턴스 혜택이 있으면 해당하는 온디맨드 인스턴스 사용량을 기준으로 다른 연결 계정에 적용됩니다.

예제

특정 리전에서 `m4.xlarge` RI 하나를 구매합니다. 이 `m4.xlarge` RI는 동일한 리전에서 모든 `m4` 인스턴스 사용에 자동으로 적용될 수 있습니다. 다음 표에서 AWS는 `m4.xlarge`를 별도의 `m4.large` 인스턴스 두 개에 적용했습니다.

<code>lineltem/LineltemType</code>	<code>RIFee</code>	<code>Discounted Usage</code>	<code>Discounted Usage</code>
<code>lineltem/ProductCode</code>	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/UNIX (Amazon VPC), m4.xlarge instance	Linux/UNIX(Amazon VPC), m4.large 예약형 인스턴스 적용됨	Linux/UNIX(Amazon VPC), m4.large 예약형 인스턴스 적용됨
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/TotalReservedNormalizedUnits	5952		

m4.large 사용량 항목 두 개는 ResourceId가 서로 다르며 둘 다 단일 m4.xlarge RI로부터 할인 혜택을 받았습니다. 이것은 사용량 및 월 기본 요금 항목 전반에 걸쳐 [reservationARN] 값을 일치시키면 표시됩니다.

RI 구매 옵션에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [청구 방식](#)을 참조하세요.

온디맨드 용량 예약 모니터링

용량 예약을 사용하면 특정 가용 영역에서 어떤 기간에 대해서든 Amazon EC2 인스턴스의 용량을 예약할 수 있습니다. 그러면 리전별 예약형 인스턴스(RI)에서 제공하는 결제 할인과 별도로 용량 예약을 생성하고 관리할 수 있습니다. 결제 할인 혜택을 누리려면 리전별 RI를 용량 예약과 함께 사용하면 됩니다.

용량 예약 항목

AWS CUR 데이터 사전에 정의된 일부 열을 사용하여 용량 예약을 추적할 수 있습니다. 다음 열은 용량 예약에도 사용됩니다.

이 섹션에서는 용량 예약과 관련된 보충 정의를 사용하여 이러한 항목을 정의합니다.

비용 및 사용 보고서 열 설명에 대한 자세한 정보는 [the section called “항목 세부 정보”](#) 섹션을 참조하세요.

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

UsageType이 Reservation 또는 DedicatedRes인 용량 예약의 경우 BlendedRate는 0입니다. 이는 용량 예약 비용이 용량 예약 자체가 아닌 용량을 제공하는 인스턴스와 관련되기 때문입니다.

R

lineItem/ResourceId

비용 및 사용 보고서를 생성할 때 lineItem/ResourceId를 포함시킨 경우 ResourceId 열을 사용하여 용량 예약을 식별하고 추적할 수 있습니다. 용량 예약 ResourceId는 UnusedBox, UnusedDed, Reservation, DedicatedRes, UsageTypes에 대해서만 캡처됩니다.

용량 예약은 항상 리소스 ID에 cr-을 포함하며 리소스 ID의 형식은 다음과 같습니다.

```
arn:aws:ec2:<region>:<account id>:<capacity-reservation>/cr-0be443example1db6f
```

U

lineItem/UnblendedCost

BlendedRate에 UsageAmount를 곱한 값과 같습니다.

lineItem/UnblendedRate

UsageType이 Reservation 또는 DedicatedRes인 용량 예약의 경우 UnblendedRate는 0입니다. 이는 용량 예약 비용이 용량 예약 자체가 아닌 용량을 제공하는 인스턴스와 관련되기 때문입니다.

lineItem/UsageAmount

사용한 용량 예약 수입입니다. 각 용량 예약에는 예약을 사용하는 인스턴스를 한 시간 동안 두 개 이상 실행할 수 있도록 한 시간에 대해 여러 개의 슬롯이 있을 수 있습니다. 따라서 한 시간에 2 이상의 인스턴스-시간을 사용할 수 있습니다. UsageAmount는 항목에 포함된 인스턴스 슬롯 수를 항목에 포함된 시간 수와 곱하여 계산됩니다.

lineItem/UsageType

사용한 특정 예약 수 Amazon EC2의 경우 옵션은 다음과 같습니다.

lineItem/lineitemtype = BoxUsage

이 UsageType의 경우 UsageAmount 열은 사용한 인스턴스의 인스턴스-시간 수입입니다.

예를 들어, 보고서에 한 시간이 포함되고 10개의 인스턴스를 포함할 수 있는 용량 예약 항목이 있습니다. 보고서에 포함된 기간 동안 2개의 인스턴스 슬롯을 사용하는 경우 BoxUsage UsageAmount에는 예약하고 사용한 인스턴스 시간 수가 포함됩니다. 이 경우 값은 2(사용한 인스턴스 슬롯 수)에 1시간(보고서에 포함된 시간)을 곱하여 총 2가 됩니다. 1일이 포함된 보고서의 경우 UsageAmount는 2에 24를 곱하여 총 48이 됩니다.

DedicatedRes

UsageType이 DedicatedRes인 경우 UsageAmount 열은 예약한 전용 용량 예약의 인스턴스-시간 수를 설명합니다.

예약

UsageType이 Reservation인 경우 UsageAmount 열은 예약한 용량 예약의 인스턴스-시간 수를 설명합니다.

예를 들어, 보고서에 한 시간이 포함되고 보고서의 용량 예약 항목에 10개의 인스턴스가 포함될 수 있는 경우 Reservation UsageAmount에는 예약한 인스턴스 슬롯 수가 포함됩니다. 이 경우 값은 10(사용 가능한 인스턴스 슬롯 수)에 1 시간(보고서에 포함된 시간)을 곱하여 총 10이 됩니다. 1일이 포함된 보고서의 경우, UsageAmount는 10에 24를 곱하여 총 240이 됩니다.

UnusedBox

UsageType이 UnusedBox인 경우 UsageAmount 열은 예약했지만 사용하지 않은 용량 예약의 인스턴스-시간 수를 설명합니다.

예를 들어, 보고서에 한 시간이 포함되고 10개의 인스턴스를 포함할 수 있는 용량 예약 항목이 있습니다. 보고서에 포함된 기간 동안 8개의 인스턴스 슬롯을 사용하지 않은 경우 UnusedBox UsageAmount에는 예약했지만 사용하지 않은 인스턴스 시간 수가 포함됩니다. 이 경우 값은 8(사용하지 않은 인스턴스 슬롯 수)에 1 시간(보고서에 포함된 시간)을 곱하여 총 8이 됩니다. 1일이 포함된 보고서의 경우 UsageAmount는 8에 24를 곱하여 총 192가 됩니다.

UnusedDed

UsageType이 UnusedDed인 경우 UsageAmount 열은 예약했지만 사용하지 않은 전용 용량 예약의 인스턴스-시간 수를 설명합니다.

데이터 전송 요금 이해

AWS CUR의 [lineItem/UsageType](#) 열을 사용하여 AWS 데이터 전송 요금을 식별할 수 있습니다.

Note

데이터 전송 요금은 사용된 서비스 및 소스 AWS 리전에 따라 달라질 수 있습니다. 자세한 요금 정보는 서비스 요금 페이지를 참조하세요. 예를 들어 Amazon EC2 데이터 전송에 대한 자세한 요금 정보는 [Amazon EC2 온디맨드 요금](#)을 참조하세요.

AWS 리전 내 데이터 전송

동일한 AWS 리전의 가용 영역 간 데이터 전송은 UsageType이 **Region-DataTransfer-Regional-Bytes**입니다. 예를 들어 USE2-DataTransfer-Regional-Bytes 사용 유형은 미국 동부(오하이오) 리전의 가용 영역 간 데이터 전송 요금을 식별합니다.

지정된 리소스에 대해 AWS 리전 내 데이터 전송의 인바운드 및 아웃바운드 트래픽 모두에 대해 요금이 부과됩니다. 즉, 측정된 각 리소스에 대해 각 데이터 전송에 대한 두 개의 DataTransfer-

Regional-Bytes 항목이 표시됩니다. 일부 서비스에서는 리전 내 트래픽이 무료로 제공되므로 자세한 정보는 서비스 요금 페이지를 확인하세요.

AWS 리전 간 데이터 전송

서로 다른 AWS 리전 간의 데이터 전송에는 다음과 같은 사용 유형이 있을 수 있습니다.

- *Source Region-Destination Region*-AWS-In-Bytes: 다른 특정 AWS 리전에서 대상 리전으로 들어오는 데이터 전송을 측정합니다.
- *Source Region-Destination Region*-AWS-Out-Bytes: 소스 리전에서 다른 특정 AWS 리전으로의 발신 데이터 전송을 측정합니다.
- *Source Region*-AWS-In-Bytes: 이 사용 유형은 트래픽이 VPC 피어링을 통해 흐를 때 나타납니다.
- *Source Region*-AWS-Out-Bytes: 이 사용 유형은 트래픽이 VPC 피어링을 통해 흐를 때 나타납니다.

각 리소스에 대해 AWS 리전 간 데이터 전송은 보고서의 두 행 항목에 해당합니다.

- 대상 리전으로 전송된 데이터의 항목
- 소스 리전에서 전송된 데이터의 항목

대상 리전으로 전송된 데이터에는 요금이 부과되지 않습니다. 데이터 전송 요금은 소스 리전에서 전송된 데이터에 따라 결정됩니다.

예를 들어, USE2 리전에서 APS3 리전으로 데이터를 전송할 때는 APS3-USE2-AWS-In-Bytes 항목과 USE2-APS3-AWS-Out-Bytes 항목이 모두 포함됩니다. APS3-USE2-AWS-In-Bytes 항목에는 해당 요금이 부과되지 않습니다. 데이터 전송 요금은 USE2-APS3-AWS-Out-Bytes 항목과 관련이 있습니다.

인터넷으로의 데이터 전송

에서 인터넷 AWS 으로의 데이터 전송에는 UsageType이 *Region*-DataTransfer-Out-Bytes. 예를 들어, USE2-DataTransfer-Out-Bytes 사용 유형은 USE2 리전에서 인터넷으로 데이터를 전송하는 데 부과되는 요금을 식별합니다.

인터넷에서 AWS로의 데이터 전송에는 요금이 부과되지 않습니다.

Note

리전 접두사가 없는 데이터 전송 사용 유형(예: `DataTransfer-Regional-Bytes` 또는 `DataTransfer-Out-Bytes`)은 미국 동부(버지니아 북부) 리전에서 데이터 전송을 나타냅니다.

Direct Connect 트래픽

Direct Connect 퍼블릭 가상 인터페이스를 통한 데이터 전송에는 `DataXfer-In` 또는 로 끝나는 사용 유형이 있습니다 `DataXfer-Out`.

Direct Connect 프라이빗 또는 전송 가상 인터페이스를 통한 데이터 전송에는 `DataXfer-In:dc.3` 또는 로 끝나는 사용 유형이 있습니다 `DataXfer-Out:dc.3`.

S3 전송 가속 트래픽

S3 전송 가속을 사용한 Amazon S3 데이터 전송에는 `ABytes`를 포함하는 사용 유형이 있습니다.

- Amazon S3와 Amazon EC2 간: `C3DataTransfer-In-ABytes` 또는 `C3DataTransfer-Out-ABytes`로 끝나는 사용 유형
- Amazon S3와 인터넷 간: `DataTransfer-In-ABytes` 또는 `DataTransfer-Out-ABytes`로 끝나는 사용 유형
- Amazon S3와 CloudFront 간: `CloudFront-In-ABytes` 또는 `CloudFront-Out-ABytes`로 끝나는 사용 유형
- 서로 다른 AWS 리전의 Amazon S3 버킷 간:의 사용 유형 *Source Region-Destination Region-AWS-Out-ABytes*

CloudFront 트래픽

CloudFront 데이터 전송의 사용 유형은 *Region-DataTransfer-Out-Bytes* 또는 `AmazonCloudFront` 제품 코드와 결합된 *Region-DataTransfer-Out-0Bytes*입니다. 사용 유형의 리전 접두사는 데이터 전송에 사용된 CloudFront 엣지 로케이션을 나타냅니다. 예를 들어, `AP-DataTransfer-Out-Bytes` 사용 유형은 AP 리전에서 인터넷으로 데이터를 전송하는 데 부과되는 요금을 식별합니다.

i Tip

[lineitem/ProductCode](#) 열을 사용하여 CloudFront 데이터 전송과 인터넷으로의 데이터 전송을 구분할 수 있습니다. 이러한 데이터 전송 유형의 사용 유형은 비슷해 보입니다.

분할 비용 할당 데이터에 대한 이해

비용 및 사용 보고서(AWS CUR)를 사용하여 Amazon ECS 및 Amazon EKS 컨테이너 비용을 추적할 수 있습니다. 분할 비용 할당 데이터를 사용하면 컨테이너 워크로드가 공유 컴퓨팅 및 메모리 리소스를 사용하는 방식에 따라 개별 사업부와 팀에 컨테이너 비용을 할당할 수 있습니다. 비용 할당 데이터 분할은 새로운 컨테이너 수준 리소스(즉, ECS 작업 및 Kubernetes 포드)에 대한 비용 및 사용량 데이터를 AWS CUR에 도입합니다. 이전에는 AWS CUR이 EC2 인스턴스 수준에서만 비용을 지원했습니다. 분할 비용 할당 데이터는 각 컨테이너의 EC2 인스턴스 리소스 사용량을 살펴봄으로써 컨테이너 수준 비용을 생성하고, 인스턴스의 분할 상한 비용과 인스턴스에서 실행된 컨테이너가 사용한 CPU 및 메모리 리소스의 비율을 기반으로 비용을 생성합니다.

Amazon EKS와 함께 사용되는 가속 컴퓨팅 인스턴스의 경우 분할 비용 할당 데이터에는 CPU 및 메모리와 함께 특수 프로세서에 대한 리소스 할당이 포함됩니다. 여기에는 NVIDIA 및 AMD GPUs, AWS Trainium 및 AWS Inferentia 액셀러레이터가 포함됩니다. 이 기능은 Amazon EKS 환경에서만 사용할 수 있으며 이러한 가속화된 컴퓨팅 리소스에 대한 포드 수준 리소스 예약 데이터를 제공합니다. 이를 통해 AI/ML 애플리케이션 및 기타 컴퓨팅 집약적인 작업과 같이 이러한 특수 프로세서를 사용하는 워크로드에 대한 비용을 추적하고 할당할 수 있습니다. 현재 가속 컴퓨팅 인스턴스 목록은 [가속 컴퓨팅](#)을 참조하세요.

분할 비용 할당 데이터에는 AWS CUR의 컨테이너화된 각 리소스 ID(즉, ECS 작업 및 Kubernetes 포드)에 대한 새 사용 레코드와 새 비용 지표 열이 도입됩니다. 자세한 내용은 [분할 항목 세부 정보](#)를 참조하세요.

AWS CUR에 분할 비용 할당 데이터를 포함하면 CPU 및 메모리 비용을 반영하기 위해 시간당 각 ECS 작업 및 Kubernetes 포드에 대해 두 개의 새 사용 레코드가 추가됩니다. 일일 AWS CUR의 새 행 항목 수를 추정하려면 다음 공식을 사용합니다.

ECS: $(\text{number of tasks} * \text{average task lifetime} * 2) * 24$

EKS: $(\text{number of pods} * \text{average pod lifetime} * 2) * 24$

예를 들어, 10개의 EC2 인스턴스로 구성된 클러스터에서 매시간 1,000개의 포드가 실행되고 있고 포드 수명이 1시간 미만인 경우 다음과 같습니다.

$(1000 * 1 * 2) * 24 = 48,000$ new usage records in AWS CUR

Amazon EKS의 가속 컴퓨팅 인스턴스의 경우 액셀러레이터, CPU 및 메모리 비용을 반영하기 위해 시간당 각 Kubernetes 포드에 대해 3개의 새 사용 레코드가 추가됩니다. 일일 AWS CUR의 새 행 항목 수를 추정하려면 다음 공식을 사용합니다.

가속 컴퓨팅을 사용하는 EKS의 경우: $(\text{number of pods} * \text{average pod lifetime} * 3) * 24$

예를 들어, 10개의 EC2 인스턴스로 구성된 클러스터에서 매시간 1,000개의 포드가 실행되고 있고 각 포드 수명이 1시간 미만인 경우 다음과 같습니다. $(1000 * 1 * 3) * 24 = 72,000$ new usage records in AWS CUR

Note

ECS: AWS 비용 할당 태그의 경우 비용 및 사용 보고서에 Amazon ECS 관리형 태그 또는 사용자 추가 태그를 사용할 수 있습니다. 이러한 태그는 모든 새로운 ECS 비용 할당 데이터 분할 사용 기록에 적용됩니다. 자세한 정보는 [결제를 위한 ECS 리소스 태그 지정](#)을 참조하세요.

EKS: 비용 할당 데이터 분할은 일부 Kubernetes 특성에 대한 새 비용 할당 태그를 생성합니다. 해당 태그에는 `aws:eks:cluster-name`, `aws:eks:deployment`, `aws:eks:namespace`, `aws:eks:node`, `aws:eks:workload-name` 및 `aws:eks:workload-type`이 포함됩니다.

- `aws:eks:cluster-name`, `aws:eks:namespace` 및 `aws:eks:node`는 각각 클러스터, 네임스페이스 및 노드의 이름으로 채워집니다.
- `aws:eks:workload-type`은 포드를 관리하는 워크로드가 정확히 하나이며 기본 제공 워크로드 중 하나일 경우에만 채워집니다. 워크로드 유형은 ReplicaSet, StatefulSet, Job, DaemonSet 또는 ReplicationController를 포함하여 `aws:eks:workload-name`에는 워크로드 이름을 포함합니다. 자세한 내용은 Kubernetes 문서의 [워크로드](#)를 참조하세요.
- `aws:eks:deployment`는 워크로드 유형 ReplicaSet에 대해서만 채워집니다. 이는 ReplicaSet를 생성하는 배포입니다.

이러한 태그는 모든 새로운 EKS 비용 할당 데이터 분할 사용 기록에 적용됩니다. 이러한 태그는 기본적으로 비용 할당에 대해 활성화됩니다. 이전에 `aws:eks:cluster-name` 태그를 사용하고 비활성화한 경우 비용 할당 데이터 분할은 이 설정을 유지하고 태그를 활성화하지 않습니다. [비용 할당 태그](#) 콘솔 페이지에서 활성화할 수 있습니다.

비용 할당 데이터 분할 활성화

Note

비용 할당 데이터 분할은 Cost Explorer에서는 사용할 수 없습니다. 데이터 내보내기를 사용하는 레거시 비용 및 사용량 보고서(CUR)와 비용 및 사용량 보고서 2.0(CUR 2.0)에서 사용할 수 있습니다.

비용 관리 기본 설정을 통해 비용 할당 데이터를 분할하도록 선택하는 것은 전제 조건입니다.

비용 할당 데이터 분할 선택

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
 2. 탐색 창에서 비용 관리 기본 설정을 선택합니다.
 3. 일반의 비용 할당 데이터 분할 섹션에서 다음 중 하나를 선택합니다.
 - Amazon Elastic Container Service(Amazon ECS)는 Amazon ECS에만 옵트인합니다.
 - Amazon Elastic Kubernetes Service(Amazon EKS)는 Amazon EKS에만 옵트인합니다. Amazon EKS의 경우 다음 중 하나를 선택합니다.
 - 리소스 요청: Kubernetes 포드 CPU 및 메모리 리소스별로만 Amazon EC2를 할당합니다. 이를 통해 애플리케이션 팀이 필요한 것만 프로비저닝할 수 있습니다.
 - Amazon Managed Service for Prometheus: Kubernetes 포드 CPU 및 메모리 리소스 요청과 실제 사용률 중 더 높은 값을 기준으로 Amazon EC2 비용을 할당합니다. 이를 통해 각 애플리케이션 팀이 사용한 비용을 지불할 수 있습니다. Amazon Managed Service for Prometheus를 설정하는 방법에 대한 자세한 내용은 Amazon Managed Service for Prometheus 사용 설명서의 [설정](#)을 참조하세요.
- 사전 조건:의 모든 기능을 활성화해야 합니다 AWS Organizations. 자세한 내용은 Organizations 사용자 가이드의 [조직 내 모든 기능 활성화](#)를 참조하세요.
- Amazon CloudWatch Container Insights: 이 기능은 공유 EC2 인스턴스를 사용하여 여러 애플리케이션 컨테이너를 실행하는 클러스터에 대해 보다 세분화된 비용 가시성을 제공하여, EKS 클러스터의 공유 비용에 대한 더 나은 비용 할당을 가능하게 합니다.

Note

- 일반 및 지급인 계정만 AWS Cost Management 기본 설정에 액세스할 수 있으며 비용 할당 데이터를 분할하도록 옵트인할 수 있습니다. 선택하면 멤버 계정은 비용 및 사용 보고서에서 데이터를 볼 수 있습니다.
- 리소스 요청을 선택할 경우 메모리 및 CPU 요청으로 구성된 포드만 비용 할당 데이터 분할에 사용됩니다. 사용량을 요청하지 않은 포드에는 비용 데이터 분할이 표시되지 않습니다.
- Amazon Managed Service for Prometheus를 선택하는 경우 AWS Organizations의 모든 기능을 활성화해야 합니다. 자세한 내용은 [조직 내 모든 기능 활성화](#)를 참조하세요. 또한 비용 할당 데이터 분할은 비용 할당 데이터 분할에 의해 사용되거나 관리되는 AWS 서비스 및 리소스에 액세스할 수 있는 새로운 서비스 연결 역할을 생성합니다.
- 가속 컴퓨팅 인스턴스의 경우 리소스 요청 옵션만 지원됩니다. 이러한 인스턴스에는 Amazon Managed Service for Prometheus 또는 Amazon CloudWatch Container Insights가 지원되지 않습니다. 가속 컴퓨팅 인스턴스를 사용하는 경우 다른 측정 옵션이 활성화된 경우에도 시스템은 기본적으로 액셀러레이터, CPU 및 메모리 비용을 계산하기 위한 리소스 요청으로 설정됩니다.

선택하면 보고서 생성의 1단계에서 또는 이후 보고서 세부 정보를 편집하여 컨테이너 수준 리소스의 비용 및 사용 데이터를 보고서에 포함하도록 선택할 수 있습니다.

보고서에 비용 및 사용량 데이터 포함

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. 새 보고서를 만들든 기존 보고서를 수정하든, 보고서 세부 정보 지정 페이지의 보고서 내용에서 비용 할당 데이터 분할을 선택합니다.

Note

AWS CUR API 또는 AWS Command Line Interface (CLI)를 사용하여 분할 비용 할당 데이터 기본 설정을 관리할 수도 있습니다.

비용 할당 데이터 분할을 사용하면 전체 통합 결제 패밀리(지급인 및 연결 계정)의 모든 Amazon ECS 및 Amazon EKS 컨테이너 객체에 대한 비용을 파악할 수 있습니다. 비용 할당 데이터 분할이 활성화되

면 작업 및 컨테이너를 자동으로 스캔합니다. 컨테이너 워크로드에 대한 원격 측정 사용 데이터를 수집하고 이번 달의 세분화된 비용 데이터를 준비합니다.

Note

데이터가 CUR에 표시되는 데 최대 AWS 24시간이 걸릴 수 있습니다.

결제 및 비용 관리 페이지의 액세스 관리에 대한 자세한 정보는 [액세스 권한 관리 개요](#)를 참조하세요.

AWS Cost Management 기본 설정 및 Cost Explorer에 대한 액세스 제어에 대한 자세한 내용은 [Cost Explorer에 대한 액세스 제어](#)를 참조하세요.

비용 할당 데이터 분할의 예제

다음 예제의 목적은 개별 Amazon ECS 서비스, Amazon ECS 클러스터의 작업, Amazon EKS 클러스터의 Kubernetes 네임스페이스 및 포드의 비용을 계산하여 비용 할당 데이터 분할을 계산하는 방법을 보여주는 것입니다. 예제 전체에서 사용된 요금은 설명 목적으로만 사용됩니다.

Note

이 예제에서는 Amazon EKS 클러스터에서 실행되는 Kubernetes 네임스페이스 및 포드를 설명합니다. 그런 다음, Amazon ECS 클러스터에서 실행되는 Amazon ECS 서비스 및 작업에 동일한 비용 모델을 적용할 수 있습니다.

한 시간 동안 다음과 같은 사용량이 발생합니다.

- 한 시간 동안 실행되는 네임스페이스 2개와 포드 4개가 포함된 단일 인스턴스(m5.xlarge) 공유 클러스터.
- 인스턴스 구성은 4개의 vCPU와 16GB의 메모리입니다.
- 인스턴스의 분할 상한 비용은 시간당 1 USD입니다.

비용 할당 데이터 분할은 9:1 비율을 기준으로 CPU와 메모리의 상대적 단위 가중치를 사용합니다. 이는 [AWS Fargate](#)의 시간당 vCPU별 및 시간당 GB별 요금에서 파생됩니다.

1단계: CPU 및 메모리의 단가 계산

$$\text{Unit-cost-per-resource} = \text{Hourly-instance-cost} / ((\text{Memory-weight} * \text{Memory-available}) + (\text{CPU-weight} * \text{CPU-available}))$$

$$= 1 \text{ USD} / ((1 * 16\text{GB}) + (9 * 4\text{vCPU})) = 0.02 \text{ USD}$$

$$\text{Cost-per-vCPU-hour} = \text{CPU-weight} * \text{Unit-cost-per-resource}$$

$$= 9 * 0.02 \text{ USD} = 0.17 \text{ USD}$$


$$\text{Cost-per-GB-hour} = \text{Memory-weight} * \text{Unit-cost-per-resource}$$

$$= 1 * 0.02 \text{ USD} = 0.02 \text{ USD}$$

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

2단계: 할당된 용량 및 인스턴스 미사용 용량 계산

- 할당된 용량: 상위 EC2 인스턴스에서 Kubernetes 포드에 할당된 메모리 및 vCPU로, 최대 사용된 용량과 예약 용량으로 정의됩니다.

 Note

메모리 또는 vCPU 사용 데이터를 사용할 수 없는 경우 예약 데이터가 대신 사용됩니다. 자세한 내용은 [Amazon ECS 사용 보고서](#) 또는 [Amazon EKS 비용 모니터링](#)을 참조하세요.

- 인스턴스 미사용 용량: vCPU 및 메모리의 미사용 용량입니다.

$$\text{Pod1-Allocated-vCPU} = \text{Max} (1 \text{ vCPU}, 0.1 \text{ vCPU}) = 1 \text{ vCPU}$$

$$\text{Pod1-Allocated-memory} = \text{Max} (4 \text{ GB}, 3 \text{ GB}) = 4\text{GB}$$

$$\text{Instance-Unused-vCPU} = \text{Max} (\text{CPU-available} - \text{SUM}(\text{Allocated-vCPU}), 0) = \text{최대}(4 - 4.9, 0) = 0$$

Instance-Unused-memory = Max (Memory-available - SUM(Allocated-memory), 0)
 = 최대(16 - 14, 0) = 2GB

이 예제에서 인스턴스는 예약된 것보다 더 많은 vCPU를 사용하는 Pod2에 따른 CPU 초과 구독을 사용하고 있습니다.

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

3단계: 분할 사용률 계산

- 분할 사용률: EC2 인스턴스에서 사용 가능한 전체 CPU 또는 메모리 대비 Kubernetes 포드에 사용된 CPU 또는 메모리의 비율입니다.
- 미사용 비율: EC2 인스턴스에서 사용된 전체 CPU 또는 메모리 대비 Kubernetes 포드에 사용된 CPU 또는 메모리의 백분율(즉, 인스턴스의 미사용 CPU 또는 메모리를 고려하지 않음)입니다.

Pod1-vCPU-split-usage-ratio = Allocated-vCPU / Total-vCPU

= 1 vCPU/4.9vCPU = 0.204

Pod1-Memory-split-usage-ratio = Allocated-GB / Total-GB

= 4GB/16GB = 0.250

Pod1-vCPU-unused-ratio = Pod1-vCPU-split-usage-ratio / (Total-CPU-split-usage-ratio - Instance-unused-CPU)(인스턴스 미사용 CPU가 0인 경우 0으로 설정됨)

=0(인스턴스 미사용 CPU가 0이므로)

Pod1-Memory-unused-ratio = Pod1-Memory-split-usage-ratio / (Total-Memory-split-usage-ratio - Instance-unused-memory)(인스턴스 미사용 메모리가 0인 경우 0으로 설정됨)

= 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	

4단계: 분할 비용 및 미사용 비용 계산

- 분할 비용: Kubernetes 포드에 할당된 CPU 및 메모리 사용량을 기준으로 EC2 인스턴스 비용을 할당하는 용도입니다.
- 미사용 인스턴스 비용: 인스턴스에서 미사용 CPU 또는 메모리 리소스의 비용입니다.

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 USD) + (0.25 * 16GB * 0.02 USD) = 0.22 USD

$$\text{Pod1-Unused-cost} = (\text{Pod1-vCPU-unused-ratio} * \text{Instance-vCPU-unused-ratio} * \text{vCPU-available} * \text{Cost-per-VCPU-hour}) + (\text{Pod1-Memory-unused-ratio} * \text{Instance-Memory-unused-ratio} * \text{Memory-available} * \text{Cost-per-GB-hour})$$

$$= (0 * 0 * 4 * 0.17 \text{ USD}) + (0.286 * 0.125 * 16 * 0.02 \text{ USD}) = 0.01 \text{ USD}$$

$$\text{Pod1-Total-split-cost} = \text{Pod1-Split-cost} + \text{Pod1-Unused-cost}$$

$$= 0.23 \text{ USD}$$

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

서비스 비용은 각 네임스페이스와 관련된 포드 비용의 합계입니다.

$$\text{Namespace1} = 0.23 \text{ USD} + 0.19 \text{ USD} = 0.42 \text{ USD}$$

$$\text{Namespace2} = 0.40 \text{ USD} + 0.19 \text{ USD} = 0.59 \text{ USD}$$

샘플 AWS CUR

청구 기간 동안 EC2 인스턴스의 전체 사용량을 보장하는 절감형 플랜을 사용하는 경우 분할 상한 비용은 savingsPlan/SavingsPlanEffectiveCost를 사용하여 계산됩니다.

lineltem/ResourceID	lineltem/lineltemType	lineltem/UsageType	lineltem/UnblendedCost	lineltem/NetUnblendedCost	savingsPlan/SavingsPlanEffectiveCost	savingsPlan/NetSavingsPlanEffectiveCost	splitLineltem/ParentResourceId	splitLineltem/SplitUsage	splitLineltem/SplitCost	splitLineltem/NetSplitCost	splitLineltem/UnusedCost	splitLineltem/NetUnusedCost
i-12345	SavingsPlanCoveredUsage	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

청구 기간 동안 EC2 인스턴스의 일부 사용량을 보장하는 절감형 플랜을 사용하고 나머지 EC2 인스턴스 사용에 대해 온디맨드 요금으로 청구되는 경우, EC2 인스턴스 분할 상한 비용은 savingsPlan/SavingsPlanEffectiveCost(SavingsPlanCoveredUsage의 경우) + lineltem/UnblendedCost(온디맨드 사용의 경우)를 사용하여 계산됩니다.

lineltem/ResourceID	lineltem/lineltemType	lineltem/UsageType	lineltem/UnblendedCost	lineltem/NetUnblendedCost	savingsPlan/SavingsPlanEffectiveCost	savingsPlan/NetSavingsPlanEffectiveCost	splitLineltem/ParentResourceId	splitLineltem/SplitUsage	splitLineltem/SplitCost	splitLineltem/NetSplitCost	splitLineltem/UnusedCost	splitLineltem/NetUnusedCost
i-12345	SavingsPlanCoveredUsage	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

가속화된 인스턴스에 대한 분할 비용 할당 데이터의 예제

다음 예제의 목적은 Amazon EKS 클러스터의 Kubernetes 네임스페이스 및 포드의 비용을 계산하여 분할 비용 할당 데이터를 계산하는 방법을 보여주는 것입니다. 예제 전체에서 사용된 요금은 설명 목적으로만 사용됩니다.

한 시간 동안 다음과 같은 사용량이 발생합니다.

- 두 네임스페이스에서 포드 4개를 실행 중이며 각 네임스페이스의 비용을 이해하고자 하는 단일 EC2 인스턴스입니다.
- EC2 인스턴스는 8개의 GPU, 64개의 vCPU 및 488GB RAM이 있는 p3.16xlarge입니다.
- 인스턴스의 분할 상한 비용은 시간당 10 USD입니다.

분할 비용 할당 데이터는 9:1의 GPU:(cpu: 메모리) 상대 비율을 기반으로 리소스당 비용을 정규화합니다. 즉, GPU 단위의 비용은 CPU 및 메모리 단위의 9배입니다. 그런 다음 CPU와 메모리에 9:1의 가중

치가 할당됩니다. 가속화되지 않은 EC2 인스턴스의 경우, 현재 기본 동작이 채택되며 cpu: 메모리 가중치 기본값은 9:1입니다.

1단계: 단위 비용 계산

EC2 인스턴스의 cpu 및 메모리 리소스를 기반으로 하고 위에서 언급한 비율을 사용하여, 분할 비용 할당 데이터는 먼저 GPU당, vCPU-hr 및 GB-hr 단위 비용을 계산합니다.

GPU-Weight =9

GPU+Memory-Weight =1

CPU-Weight=1*.9=.9

Memory-Weight=1*0.1=0.1

Hourly-Instance-Cost=\$10

GPU-Available=8

Memory-Available=488

CPU-Available=64

$$\text{UnitCostPerResource} = \text{Hourly-Instance-Cost} / ((\text{GPU-Weight} * \text{GPU-Available}) + (\text{Memory-Weight} * \text{Memory-Available}) + (\text{CPU-Weight} * \text{CPU-Available})) = \$10 / ((9*8\text{gpu}) + (0.1 * 488\text{GB}) + (.9 * 64\text{vcpu})) = \$0.056$$

$$\text{Cost-per-GPU-Hour} = \text{GPU-Weight} * \text{UnitCostPerResource} = 9 * \$0.056 = \$0.504$$

$$\text{Cost-per-vcpu-Hour} = \text{CPU-Weight} * \text{UnitCostPerResource} = .9 * \$0.056 = \$0.05$$

$$\text{Cost-per-GB-Hour} = \text{Memory-Weight} * \text{UnitCostPerResource} = .1 * \$0.056 = \$0.00506$$

표 1: 단위 비용 계산

Instance	인스턴스 유형	vCPU 사용 가능	GPU 사용 가능	**	사용할 수 있는 메모리	시간당 분할 상환 비용	vCPU 시간당 비용	GPU 시간당 비용	GB 시간당 비용
인스턴스 1	p3.16xlarge	64	8		488	10 USD	0.05 USD	0.50 USD	0.005

2단계: 할당 및 미사용 용량 계산

할당된 용량

상위 EC2 인스턴스에서 Kubernetes 포드에 할당된 GPU, vCPU 및 메모리는 예약 용량, 사용된 용량 중 최대값으로 정의됩니다.

인스턴스 미사용 용량

GPU, vcpu 및 메모리의 미사용 용량

$$\text{Pod1-Allocated-GPU} = \text{Max} (1 \text{ GPU}, 1 \text{ GPU}) = 1 \text{ GPU}$$

$$\text{Pod1-Allocated-vcpu} = \text{Max} (16 \text{ vcpu}, 4 \text{ vcpu}) = 16 \text{ vcpu}$$

$$\text{Pod1-Allocated-Memory} = \text{Max} (100 \text{ GB}, 60 \text{ GB}) = 100 \text{ GB}$$

$$\begin{aligned} \text{Instance-Unused-GPU} &= \text{Max} (\text{GPU-Available} - \text{SUM}(\text{Allocated-vcpu}), 0) \\ &= \text{Max} (8 - 8, 0) = 0 \end{aligned}$$

$$\begin{aligned} \text{Instance-Unused-vcpu} &= \text{Max} (\text{CPU-Available} - \text{SUM}(\text{Allocated-vcpu}), 0) \\ &= \text{Max} (16 - 18, 0) = 0 \end{aligned}$$

$$\begin{aligned} \text{Instance-Unused-Memory} &= \text{Max} (\text{Memory-Available} - \text{SUM}(\text{Allocated-Memory}), 0) \\ &= \text{Max} (488 - 440, 0) = 48 \text{ GB} \end{aligned}$$

이 예제에서 인스턴스는 예약된 양보다 더 많은 GPU와 vCPU를 사용한 포트 2로 인해 CPU 초과 구독을 사용하고 있습니다.

표 2: 할당 및 미사용 용량 계산

포트 이름	네임스페이스	vcpu 예약	vcpu 사용	vcpu 할당	GPU 예약	GPU 사용	GPU 할당	메모리 예약	메모리 사용	메모리 할당
포트 1	네임스페이스 1	16	4	16	1	1	1	100	60	100
포트 2	네임스페이스 2	16	18	18	2	3	3	100	140	140
포트 3	네임스페이스 1	16	4	16	2	1	2	100	60	100
포트 4	네임스페이스 2	16	4	16	2	2	2	100	40	100
미사용	미사용	0	34	0	1	1	0	88	188	48
***		64	32	66	8	8	8	488	488	488

3단계: 분할 사용률 및 사용률 계산

분할 사용률

EC2 인스턴스에서 사용 가능한 전체 CPU 또는 메모리 대비 Kubernetes 포드에 사용된 CPU 또는 메모리의 비율입니다.

미사용 비율

EC2 인스턴스에서 사용된 전체 CPU 또는 메모리 대비 Kubernetes 포드에 사용된 CPU 또는 메모리의 백분율(즉, 인스턴스의 미사용 CPU 또는 메모리를 고려하지 않음)입니다.

EC2 인스턴스에서 사용 가능한 전체 CPU 또는 메모리 대비 Kubernetes 포드에 사용된 CPU 또는 메모리의 비율입니다.

$$\text{Pod1-GPU-Utilization-Ratio} = \text{Allocated-GPU} / \text{Total-GPU}$$

$$= 1 \text{ gpu} / 8 \text{ gpu} = 0.125$$

$$\text{Pod1-vcpu-Utilization-Ratio} = \text{Allocated-vcpu} / \text{Total-vcpu}$$

$$= 16 \text{ vcpu} / 66 \text{ vcpu} = 0.24$$

$$\text{Pod1-Memory-Utilization-Ratio} = \text{Allocated-GB} / \text{Total-GB}$$

$$= 100 \text{ GB} / 488\text{GB} = 0.205$$

$$\text{Pod1-GPU-Split-Ratio} = \text{Pod1-GPU-Utilization-Ratio} / (\text{Total-GPU-Utilization-Ratio} - \text{Instance-Unused-GPU}). \text{ Set to } 0 \text{ if Instance-Unused-GPU} = 0$$

$$= 0 \text{ since Instance-Unused-GPU is } 0$$

$$\text{Pod1-vcpu-Split-Ratio} = \text{Pod1-CPU-Utilization-Ratio} / (\text{Total-CPU-Utilization-Ratio} - \text{Instance-Unused-CPU}). \text{ Set to } 0 \text{ if Instance-Unused-CPU} = 0$$

$$= 0 \text{ since Instance-Unused-CPU is } 0$$

$$\text{Pod1-Memory-Split-Ratio} = \text{Pod-Memory-Utilization-Ratio} / (\text{Total-Utilization-Ratio} - \text{Instance-Unused-Memory}). \text{ Set to } 0 \text{ if Instance-Unused-Memory} = 0$$

$$= 0.204 / (1 - 0.102) = 0.227$$

표 3: 사용률 계산

포드 이름	네임스페이스	vcpu 사용률	vcpu 분할 비율	GPU 사용률	GPU 분할 비율	메모리 사용률	메모리 분할 비율
포드 1	네임스페이스 1	0.242	0	0.125	0	0.205	0.227
포드 2	네임스페이스 2	0.277	0	0.375	0	0.287	0.318
포드 3	네임스페이스 1	0.242	0	0.25	0	0.205	0.227
포드 4	네임스페이스 2	0.242	0	0.25	0	0.205	0.227
미사용	미사용	0				0.098	
		1	0	1	0	1	1

4단계: 분할 비용 및 미사용 비용 계산

비용 분할

Kubernetes 포드에 할당된 CPU 및 메모리 사용량을 기준으로 EC2 인스턴스 비용을 할당하는 용도입니다.

미사용 인스턴스 비용

인스턴스에서 미사용 CPU 또는 메모리 리소스의 비용입니다.

$$\text{Pod1-Split-Cost} = (\text{Pod1-GPU-Utilization-Ratio} * \text{GPU-Available} * \text{Cost per GPU-Hour}) + (\text{Pod1-vcpu-Utilization-Ratio} * \text{vcpu-Available} * \text{Cost per vcpu-Hour}) + (\text{Pod1-Memory-Utilization-Ratio} * \text{Memory-Available} * \text{Cost per GB-Hour})$$

$$= (.125 * 8 \text{gpu} * \$0.504) + (0.242 * 64 \text{ vcpu} * \$0.05) + (0.204 * 488 \text{GB} * \$0.00506) = 0.504 + 0.774 + 0.503 = \$1.85$$

$$\text{Pod1-Unused-Cost} = (\text{GPU-Split-Ratio} * \text{Unused-Cost}) + (\text{vcpu-Split-Ratio} * \text{Unused-Cost}) + (\text{Memory-Split-Ratio} * \text{Unused-Cost})$$

$$= (0 * 0 * 8 * \$0.504) + (0 * \$0.05) + (0.227 * .102 * 488\text{GB} * \$0.00506) = \$0.06$$

$$\text{Pod1-Total-Split-Cost} = \text{Pod1-Split-Cost} + \text{Pod1-Unused-Cost} = \$1.85 + \$0.06 = \$1.91$$

[참고: 미사용 비용 = 미사용 사용률 * 총 리소스 * 리소스 시간당 비용]

표 4 - 클러스터 내에서 실행되는 모든 포드에 대해 시간당 계산된 분할 및 미사용 비용 요약

포드 이름	네임스페이스	비용 분할	미사용 비용	총 비용
포드 1	네임스페이스 1	1.85 USD	0.06 USD	1.91 USD
포드 2	네임스페이스 2	3.18 USD	0.09 USD	3.26 USD
포드 3	네임스페이스 1	2.35 USD	0.06 USD	2.41 USD
포드 4	네임스페이스 2	2.35 USD	0.06 USD	2.41 USD
합계				10 USD

EKS에서 비용 할당에 Kubernetes 레이블 사용

분할 비용 할당 데이터는 Amazon EKS 클러스터의 비용 할당 태그로 Kubernetes 레이블을 지원합니다. 이러한 레이블은 사용자 정의 비용 할당 태그로 자동으로 가져오지만 관리 계정 수준에서 활성화해야 합니다. 활성화되면 이를 사용하여 비용 센터, 애플리케이션, 사업부 및 환경과 같은 사용자 지정 속성을 사용하여 비용 및 사용 보고서(CUR)에서 포드 수준 비용을 속성화할 수 있습니다.

이 기능은 조직이 팀, 프로젝트 또는 부서 간에 공유 EKS 환경의 비용을 정확하게 추적하고 할당하는 데 도움이 됩니다. Kubernetes 레이블을 사용하면 특정 비즈니스 요구 사항 및 조직 설계에 따라 Kubernetes 비용을 할당할 수 있습니다.

사전 조건

분할 비용 할당 데이터와 함께 Kubernetes 레이블을 사용하기 위한 사전 조건:

- AWS Billing and Cost Management 콘솔에서 분할 비용 할당 데이터를 활성화해야 합니다. 관리 계정 수준에서 활성화해야 합니다. 자세한 내용은 [Enabling split cost allocation data](#)를 참조하세요.

- 비용 할당 데이터 분할을 추적하려는 EKS 클러스터가 필요합니다. 기존 클러스터이거나 새로 생성할 수 있습니다. 자세한 내용은 Amazon EKS 사용 설명서의 [Amazon EKS 클러스터 생성](#)을 참조하세요.
- EKS 클러스터의 포드에 레이블이 할당되어 있어야 합니다. Kubernetes에서 레이블을 생성하는 방법에 대한 자세한 내용은 Kubernetes 설명서의 [레이블 및 선택기](#)를 참조하세요.

EKS에서 Kubernetes 레이블 작업

분할 비용 할당 데이터는 포드당 최대 50개의 Kubernetes 레이블을 지원하며, 비용 할당 태그로 가져오기 전에 사전순으로 정렬됩니다. 처음 50개를 초과하는 모든 레이블은 자동으로 삭제됩니다. 50개 레이블 수 제한에 도달한 후 새 비용 할당 태그를 추가해야 하는 경우 먼저 기존 레이블을 제거하고 알파벳순으로 정렬할 때 새 레이블이 처음 50개 이내인지 확인해야 합니다.

Note

일부 AWS 관리형 서비스는 EKS 포드에 레이블을 자동으로 추가합니다. 이러한 레이블은 포드당 50개 레이블 수 제한에 포함되며 비용 할당 태그 페이지에 표시됩니다. Kubernetes 레이블에는 크기 제한이 없지만, 비용 할당 태그에는 태그 키는 128자, 태그 값은 256자의 구체적인 글자 수 제한이 있습니다. 이러한 글자 수 제한을 초과하는 레이블은 삭제되며 비용 할당 태그로 표시되지 않습니다. 비용 할당을 위해 이러한 글자 수 제한을 따르는 레이블을 생성하는 것이 좋습니다.

가져온 Kubernetes 레이블은 비용 할당 태그로 표시되며 지급인 계정 수준에서 활성화해야 합니다. 비용 할당 태그 및 활성화에 대한 자세한 내용은 [사용자 정의 비용 할당 태그 사용](#)을 참조하세요. 리소스당 사용자 정의 태그 50개와 지급인 계정당 사용자 정의 태그 500개의 비용 할당 태그 제한이 적용됩니다. 시스템 생성 태그는 이러한 제한에 포함되지 않습니다.

Note

사용자 정의 태그를 생성하여 리소스에 적용한 후 해당 태그 키가 비용 할당 태그 페이지에 나타나는 데 최대 24시간이 걸릴 수 있습니다. 태그를 활성화하면 태그가 활성화되는 데 24시간이 더 걸릴 수 있습니다.

Kubernetes 레이블 및 비용 할당 태그 관리

EKS에서 Kubernetes 레이블을 추가, 삭제 및 편집하고 관련 비용 할당 태그를 비활성화할 수 있습니다. 다음은 각 작업에 대해 예상되는 동작에 대한 설명입니다.

새 레이블 추가

포드에 새 Kubernetes 레이블을 추가할 수 있습니다. 레이블 수 제한 50개에 도달하지 않은 경우, 새 레이블이 가져와져 비용 할당 태그로 제공되며, 이후 활성화할 수 있습니다. 그러나 50개 제한에 도달하면 처음 50개 레이블의 알파벳 정렬 순서 내에 있더라도 새 레이블을 가져오지 않습니다. 새 레이블을 가져오려면 먼저 기존 비용 할당 태그를 비활성화해야 합니다.

레이블 편집

Kubernetes에서는 레이블 키를 편집할 수 없습니다. 레이블 키를 변경하려면 레이블 키를 제거하고 새 레이블을 추가해야 합니다. 그러나 다음 CUR에 반영될 레이블 값을 편집할 수 있습니다.

레이블 삭제

EKS 포드에서 레이블을 제거할 수 있습니다. 레이블을 제거해도 연결된 비용 할당 태그가 자동으로 비활성화되지는 않습니다. 비용 할당 태그를 명시적으로 비활성화할 때까지 분할 비용 할당 데이터는 CUR에 계속 채워집니다.

비용 할당 태그 비활성화

Kubernetes 레이블에서 생성된 모든 비용 할당 태그를 비활성화할 수 있습니다. 비활성화되면 데이터가 해당 열에 더 이상 채워지지 않고 다음 달의 CUR에서 열이 삭제됩니다.

비용 할당을 위한 Kubernetes 레이블 관리 모범 사례

Kubernetes 레이블은 공유 비용 할당 모델링에 상당한 유연성을 제공합니다. 이 기능의 잠재력을 극대화하려면 다음 모범 사례를 따라 비용 관리 접근 방식을 최적화하는 것이 좋습니다.

레이블 수 제한 이해

label-per-pod 레이블 50개 제한은 알파벳 정렬을 기반으로 합니다. 비용 할당을 위해 알파벳순으로 정렬된 처음 50개의 레이블만 가져옵니다. 중요한 레이블이 포함되도록 레이블 이름을 신중하게 계획하여 알파벳순으로 정렬될 때 중요한 레이블이 처음 50개 내에 표시되도록 합니다.

다음 문자 제약 조건

AWS 비용 할당 태그에는 다음과 같은 문자 제한이 있습니다.

- 태그 키: 128자
- 태그 값: 256자

Kubernetes는 더 긴 레이블을 허용하지만 이러한 제한을 초과하는 레이블은 가져오지 않습니다. 성공적인 비용 할당 추적을 위해 이러한 제한 내에서 레이블을 설계합니다.

용량에 도달했을 때 새 레이블 추가

포드가 50개 레이블 수 제한에 도달하여 새 비용 할당 레이블을 추가해야 하는 경우 다음 단계를 따릅니다.

1. 기존 레이블을 검토하고 비활성화할 비용 할당 태그를 식별합니다.
2. 선택한 태그를 비활성화합니다.
3. 새 비용 할당 레이블을 추가합니다.
4. 새 레이블이 알파벳순으로 정렬된 처음 50개 레이블에 속하는지 확인합니다.

Note

비용 할당에는 처음 50개의 알파벳순으로 정렬된 레이블만 사용됩니다.

Amazon Managed Service for Prometheus에서 비용 할당 데이터 분할 사용

Amazon EKS의 비용 데이터를 분할하려면 메모리 및 CPU 사용량을 포함하여 클러스터에서 지표를 수집하고 저장해야 합니다. Amazon Managed Service for Prometheus를 이 용도로 사용할 수 있습니다.

비용 할당 데이터를 분할하도록 옵트인하고 Amazon Managed Service for Prometheus 작업 영역이 두 가지 필수 지표(container_cpu_usage_seconds_total 및 container_memory_working_set_bytes)를 수신하기 시작하면 분할 비용 할당 데이터가 지표를 자동으로 인식하고 사용합니다.

Note

두 가지 필수 지표(container_cpu_usage_seconds_total 및 container_memory_working_set_bytes)는 기본 Prometheus 스크래이프 구성과 AWS 관리형 수집기와 함께 제공되는 기본 구성에 있습니다. 그러나 해

당 구성을 사용자 지정할 경우 `container_cpu_usage_seconds_total` 및 `container_memory_working_set_bytes` 지표에서 `name`, `namespace` 및 `pod` 레이블을 다시 지정, 수정 또는 제거하지 마세요. 이러한 레이블을 다시 지정, 수정 또는 제거하면 지표 수집에 영향을 미칠 수 있습니다.

Amazon Managed Service for Prometheus를 사용하여 단일 리전의 단일 사용 계정에서 EKS 지표를 수집할 수 있습니다. Amazon Managed Service for Prometheus 작업 영역은 해당 계정 및 리전에 있어야 합니다. 비용을 모니터링하려는 각 사용 계정 및 리전에 대해 Amazon Managed Service for Prometheus 인스턴스가 하나씩 필요합니다. 동일한 사용 계정과 리전에 있는 경우에 한하여 Amazon Managed Service for Prometheus 작업 영역에서 여러 클러스터에 대한 지표를 수집할 수 있습니다.

다음 섹션에서는 EKS 클러스터에서 Amazon Managed Service for Prometheus 작업 영역으로 올바른 지표를 보내는 방법을 설명합니다.

사전 조건

비용 할당 데이터 분할과 함께 Amazon Managed Service for Prometheus를 사용하기 위한 사전 조건:

- AWS Billing and Cost Management 콘솔에서 분할 비용 할당 데이터를 활성화해야 합니다. 자세한 내용은 [Enabling split cost allocation data](#)를 참조하세요. 비용 할당 데이터를 분할하도록 선택하면 각 사용 계정에 서비스 연결 역할이 생성되어 Amazon Managed Service for Prometheus에 해당 계정의 Amazon EKS 클러스터 지표를 쿼리합니다. 자세한 내용은 [비용 할당 데이터 분할에 대한 서비스 연결 역할](#)을 참조하세요.
- 비용 할당 데이터 분할을 추적하려는 EKS 클러스터가 필요합니다. 기존 클러스터이거나 새로 생성할 수 있습니다. 자세한 내용은 Amazon EKS 사용 설명서의 [Amazon EKS 클러스터 생성](#)을 참조하세요.

Note

이후 단계에서 사용하려면 EKS `cluster ARN`, `security group IDs` 및 두 개 이상의 `subnet IDs`(여러 가용 영역에서)를 사용해야 합니다.
(선택 사항) EKS 클러스터의 인증 모드를 `API` 또는 `API_AND_CONFIG_MAP`로 설정합니다.

- EKS 클러스터와 동일한 계정 및 리전에 Amazon Managed Service for Prometheus 인스턴스가 필요합니다. 아직 없을 경우 새롭게 생성합니다. Amazon Managed Service for Prometheus 인스턴스를 생성하는 방법에 대한 자세한 내용은 Amazon Managed Service for Prometheus 사용 설명서에서 [작업 공간 생성](#)을 참조하세요.

Note

이후 단계에서 사용하려면 Amazon Managed Service for Prometheus workspace ARN이 필요합니다.

Amazon Managed Service for Prometheus에 EKS 지표 전달

EKS 클러스터와 Amazon Managed Service for Prometheus 인스턴스가 있으면 클러스터의 지표를 인스턴스로 전달할 수 있습니다. 지표는 두 가지 방법으로 보낼 수 있습니다.

- **옵션 1: [AWS 관리형 수집기를 사용합니다.](#)** 이는 EKS 클러스터에서 Amazon Managed Service for Prometheus로 지표를 보내는 가장 간단한 방법입니다. 그러나 최대 30초마다 지표를 스크레이핑하는 제한만 있습니다.
- **옵션 2: [자체 Prometheus 에이전트를 생성합니다.](#)** 이 경우 스크레이핑 구성을 정밀하게 제어할 수 있지만 생성 후 에이전트를 관리해야 합니다.

옵션 1: AWS 관리형 수집기 사용

AWS 관리형 수집기(스크레이퍼)를 사용하는 것은 EKS 클러스터에서 Amazon Managed Service for Prometheus 인스턴스로 지표를 보내는 가장 간단한 방법입니다. 다음 절차에서는 AWS 관리형 수집기를 생성하는 단계를 안내합니다. 자세한 내용은 Amazon Managed Service for Prometheus 사용 설명서의 [AWS 관리형 수집기](#)를 참조하세요.

Note

AWS 관리형 수집기의 최소 스크레이프 간격은 30초입니다. 수명이 짧은 포드가 있을 경우 스크레이퍼 간격을 15초로 설정하는 것이 좋습니다. 15초 스크레이퍼 간격을 사용하려면 옵션 2를 사용하여 [자체 Prometheus 에이전트를 생성합니다.](#)

AWS 관리형 수집기를 생성하는 세 단계는 다음과 같습니다.

1. 스크레이퍼 구성을 생성합니다.
2. 스크레이퍼를 생성합니다.
3. EKS 클러스터를 구성하여 스크레이퍼가 지표에 액세스하는 것을 허용합니다.

1단계: 스크레이퍼 구성 생성

스크레이퍼를 생성하려면 스크레이퍼 구성이 있어야 합니다. 기본 구성을 사용하거나 직접 생성할 수 있습니다. 다음은 스크레이퍼 구성을 가져오는 세 가지 방법입니다.

- AWS CLI를 사용하여 다음을 호출하여 기본 구성을 가져옵니다.

```
aws amp get-default-scraper-configuration
```

- 자체 구성을 생성합니다. 자세한 내용은 Amazon Managed Service for Prometheus 사용 설명서의 [스크레이퍼 구성](#) 지침을 참조하세요.
- Amazon Managed Service for Prometheus 사용 설명서의 동일한 [스크레이퍼 구성](#) 지침에 제공된 샘플 구성을 복사합니다.

예를 들어 스크레이퍼 구성을 편집하여 스크레이프 간격을 수정하거나 스크레이핑된 지표를 필터링할 수 있습니다.

비용 할당 데이터 분할에 필요한 두 가지만 포함하도록 스크레이핑되는 지표를 필터링하려면 다음 스크레이퍼 구성을 사용합니다.

```
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
```

스크레이퍼 구성을 마치면 2단계에서 사용할 수 있도록 base64로 인코딩해야 합니다. 구성은 텍스트 YAML 파일입니다. 파일을 인코딩하려면 <https://www.base64encode.org/>와 같은 웹사이트를 사용합니다.

2단계: 스크레이퍼 생성

이제 구성 파일이 있으므로 스크레이퍼를 생성해야 합니다. 사전 조건 섹션에 설명된 변수를 기반으로 다음 AWS CLI 명령을 사용하여 스크레이퍼를 생성합니다. EKS 클러스터의 정보를 `<EKS-CLUSTER-ARN>`, `<SG-SECURITY-GROUP-ID>` 및 `<SUBNET-ID>` 필드에 사용하고, `<BASE64-CONFIGURATION-BLOB>`을 이전 단계에서 생성한 스크레이퍼 구성으로 바꾸고, `<AMP_WORKSPACE_ARN>`을 Amazon Managed Service for Prometheus 워크스페이스 ARN으로 바꿔야 합니다.

```
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>"}
```

3단계에서 사용할 수 있도록 반환된 scraperId를 기록해 둡니다.

3단계: EKS 클러스터를 구성하여 스크레이퍼의 지표 액세스 허용

EKS 클러스터의 인증 모드가 API 또는 API_AND_CONFIG_MAP으로 설정될 경우 스크레이퍼는 올바른 클러스터 내 액세스 정책을 자동으로 보유하며 스크레이퍼는 클러스터에 액세스할 수 있습니다. 추가 구성은 필요하지 않으며 지표가 Amazon Managed Service for Prometheus로 이동해야 합니다.

EKS 클러스터의 인증 모드가 API 또는 API_AND_CONFIG_MAP로 설정되지 않을 경우 스크레이퍼가 ClusterRole 및 ClusterRoleBinding을 통해 지표에 액세스할 수 있도록 클러스터를 수동으로 구성해야 합니다. 해당 권한을 활성화하는 방법을 알아보려면 Amazon Managed Service for Prometheus 사용 설명서의 [스크레이퍼 액세스를 위한 EKS 클러스터 수동 구성](#)을 참조하세요.

스크레이퍼가 활성화되면 두 지표(container_cpu_usage_seconds_total 및 container_memory_working_set_bytes)가 모두 Amazon Managed Service for Prometheus 워크스페이스로 푸시되고 있는지 확인합니다.

```
awscurl --service="aps" --region="<REGION>" "https://aps-
workspaces.<REGION>.amazonaws.com/workspaces/<WorkSpace_ID>/api/v1/label/__name__/
values"
```

출력:

```
{
  "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"
  ]
}
```

옵션 2: 자체 Prometheus 에이전트 생성

AWS 관리형 수집기를 사용할 수 없거나 자체 Prometheus 서버가 이미 있는 경우 자체 Prometheus 인스턴스를 에이전트로 사용하여 EKS 클러스터에서 지표를 스크레이프하고 Amazon Managed Service for Prometheus로 보낼 수 있습니다.

자체 Prometheus 인스턴스를 에이전트로 사용하는 방법에 대한 자세한 지침은 Amazon Managed Service for Prometheus 사용 설명서의 [Using a Prometheus instance as a collector](#)를 참조하세요.

다음은 Prometheus 서버 스크레이프 간격과 비용 할당 데이터 분할에 필요한 컨테이너 지표를 포함하는 샘플 Prometheus 스크레이프 구성입니다. 수명이 짧은 포드가 있을 경우 기본 Prometheus 서버 스크레이프 간격을 30초에서 15초로 줄이는 것이 좋습니다. 이로 인해 Prometheus 서버 메모리 사용량이 증가할 수 있습니다.

```
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 사용 설명서의 [Helm을 사용하여 새 Prometheus 서버에서 수집 설정](#)을 따른 경우 스크레이프 구성을 업데이트할 수 있습니다.

스크레이프 구성 업데이트

1. 가이드에서 `my_prometheus_values.yaml`을 편집하고 `server` 블록에 샘플 스크레이프 구성을 포함합니다.
2. Amazon Managed Service for Prometheus 사용 설명서의 `prometheus-chart-name` 및 `prometheus-namespace`를 사용하여 다음 명령을 실행합니다.

```

helm upgrade prometheus-chart-name prometheus-community/prometheus -n prometheus-namespace -f my_prometheus_values.yaml

```

`scrape_interval` 또는 비전역 `scrape_interval`을 사용하는 방법에 대한 자세한 내용은 [Prometheus 스크레이프 구성](#)을 참조하세요.

또는 Prometheus 수신기, Prometheus 원격 쓰기 내보내기 및 AWS Sigv4 인증 확장이 있는 AWS Distro for OpenTelemetry 수집기를 사용하여 Amazon Managed Service for Prometheus에 대한 원격 쓰기 액세스 권한을 얻을 수 있습니다.

Note

AWS 관리형 수집기와 달리 Prometheus 에이전트를 설정한 후에는 에이전트를 최신 상태로 유지하고 지표를 수집하기 위해 실행해야 합니다.

Amazon Managed Service for Prometheus 비용 예측

AWS 요금 계산기를 사용하여 분할 비용 할당 데이터에 Amazon Managed Service for Prometheus를 사용하는 비용을 추정할 수 있습니다.

견적 관련 Amazon Managed Service for Prometheus 구성

1. <https://calculator.aws/> AWS 요금 계산기를 엽니다.
2. 견적 산출을 선택합니다.
3. 서비스 추가 페이지에서 검색 필드에 Amazon Managed Service for Prometheus를 입력한 다음, 구성을 선택합니다.
4. 설명 필드에 견적에 대한 설명을 입력합니다.
5. 리전을 선택합니다.
6. 인프라 세부 정보를 사용하여 비용 계산을 선택합니다. 이 옵션을 사용하면 현재 또는 제안된 인프라 설정을 기반으로 수집, 스토리지 및 쿼리 샘플 비용을 추정할 수 있습니다.
7. EC2 인스턴스 수에 전체 통합 결제 패밀리(모든 계정 및 리전 포함)에 대한 모든 클러스터의 총 EC2 인스턴스 수를 입력합니다. 를 사용하는 경우 Fargate 작업 수를 EC2 인스턴스 수에 대한 프록시로 AWS Fargate사용합니다.
8. 비용 할당 데이터를 분할하려면 `container_cpu_usage_seconds_total` 및 `container_memory_working_set_bytes`의 두 가지 지표가 필요합니다. EC2 인스턴스당 Prometheus 지표에 2를 입력합니다.
9. 비용 할당 데이터 분할의 추천 스크레이프 간격은 15초입니다. 지표 수집 간격(초)에 15를 입력합니다. 다른 간격(예: 30초)을 사용한 경우 설정한 간격으로 변경합니다.
10. 비용 할당 데이터 분할은 다른 파라미터에 특정 요구 사항을 적용하지 않으므로 비즈니스 요구 사항에 따라 나머지 입력 파라미터에 적절한 값을 입력합니다.
11. 저장 및 서비스 추가를 선택합니다.

Amazon CloudWatch Container Insights에서 분할 비용 할당 데이터 사용

Amazon EKS의 비용 데이터를 분할하려면 메모리 및 CPU 사용량을 포함하여 클러스터에서 지표를 수집하고 저장해야 합니다. Amazon CloudWatch Container Insights를 이 용도로 사용할 수 있습니다.

비용 할당 데이터를 분할하도록 옵트인하고 EKS 클러스터에서 EKS 관찰성 추가 기능을 사용하여 CloudWatch 에이전트를 설정하면 분할 비용 할당 데이터가 ContainerInsights 네임스페이스에서 두 개의 필수 지표 (pod_cpu_usage_total 및 pod_memory_working_set)를 수신하기 시작하고 자동으로 사용됩니다. EKS에 대한 전체 컨테이너 지표 세트를 보려면 Amazon CloudWatch 사용 설명서의 [Amazon EKS 및 Kubernetes Container Insights 지표](#)를 참조하세요.

다음 섹션에서는 EKS 클러스터에서 올바른 지표를 전송하여 비용 할당 데이터를 분할하는 방법을 설명합니다.

사전 조건

분할 비용 할당 데이터와 함께 Amazon CloudWatch Container Insights를 사용하기 위한 사전 조건:

- AWS Billing and Cost Management 콘솔에서 분할 비용 할당 데이터를 활성화해야 합니다. 자세한 내용은 [Enabling split cost allocation data](#)를 참조하세요.
- 비용 할당 데이터 분할을 추적하려는 EKS 클러스터가 필요합니다. 기존 클러스터이거나 새로 생성할 수 있습니다. 자세한 내용은 Amazon EKS 사용 설명서의 [Amazon EKS 클러스터 생성](#)을 참조하세요.

EKS 지표를 전달하도록 Amazon CloudWatch Container Insights 설정

EKS 지표를 전달하려면 CloudWatch 에이전트를 설정하고 구성해야 합니다. [Amazon CloudWatch Observability EKS 추가 기능 또는 Amazon CloudWatch Observability 헬름 차트](#)를 사용하여 EKS 클러스터에 CloudWatch 에이전트와 Fluent-Bit 에이전트를 설치할 수 있습니다. CloudWatch 에이전트 설치 및 설정 방법에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 [Amazon CloudWatch Observability EKS 추가 기능 설치](#)를 참조하세요.

다음은 CloudWatch 에이전트 및 EKS 추가 기능에 필요한 최소 버전입니다.

- CloudWatch 에이전트 버전: v1.300045.0
- CloudWatch Observability EKS 추가 기능 버전: v2.0.1-eksbuild.1

Amazon CloudWatch 비용 추정

분할 비용 할당 데이터와 함께 Amazon CloudWatch Container Insights를 사용하도록 기능을 활성화하면 Amazon CloudWatch Container Insights에 pod_cpu_usage_total 및 pod_memory_working_set라는 두 가지 새로운 지표가 추가됩니다. 자세한 정보는 Amazon CloudWatch 사용 설명서의 [Amazon EKS 및 Kubernetes Container Insights 지표](#)를 참조하세요.

기능과 관련된 비용을 이해하려면 다음을 수행합니다.

1. <https://aws.amazon.com/cloudwatch/pricing/>에서 Amazon CloudWatch 요금을 엽니다.
2. 유료 티어 섹션으로 이동합니다.
3. 컨테이너 인사이트 탭을 선택합니다.
4. 비용 계산에 대한 자세한 내용은 요금 예제 섹션으로 이동하여 예제 13 - Amazon EKS 및 Kubernetes용 컨테이너 인사이트를 참조하세요.

레거시 결제 보고서 이해

이 섹션에서는 및 AWS 비용 및 사용 보고서 외부에서 AWS Data Exports 제공되는 레거시 결제 보고서에 대해 설명합니다. 이 페이지는 참조용으로 제공됩니다. 하지만 나중에 이러한 보고 방법을 사용할 수 없게 되므로 AWS Data Exports 를 사용하는 것이 좋습니다.

주제

- [세부 결제 보고서 사용](#)
- [세부 결제 보고서를 비용 및 사용 보고서로 마이그레이션](#)
- [미사용 예약 비용 이해](#)
- [월별 보고서 다운로드](#)
- [월별 비용 할당 보고서 다운로드](#)
- [AWS 사용 보고서 다운로드](#)

세부 결제 보고서 사용

Important

2019년 7월 8일 현재 신규 고객은 세부 결제 보고서 기능을 사용할 수 없습니다.

세부 결제 보고서(DBR)에는 요금에 대한 AWS 비용 및 사용 보고서(AWS CUR)와 유사한 정보가 포함되어 있지만 개별 항목을 다르게 계산합니다. DBR과 AWS CUR에 모두 가입한 경우 행 항목이 일치하지 않습니다. 그러나 월말에 보고서를 최종 완성하면서 총 비용을 조정해야 합니다.

AWS 는 다음 명령 규칙을 사용하여 DBR을 Amazon S3에 CSV 파일로 저장합니다.

```
AWS account number-aws-billing-detailed-line-items-yyyy-mm.csv.zip
```

AWS 는 하루에 여러 번 세부 결제 보고서(DBR)를 다시 생성하여 보고서를 덮어씁니다. 가 보고서를 AWS 덮어쓰면 행 항목의 순서가 이전 보고서와 다를 수 있습니다. 최종 보고서는 월말에 생성됩니다. 다음 달에는가 이전 달의 최종 보고서를 덮어쓰는 대신 새 보고서 파일을 AWS 생성합니다. 이전 달 보고서는 삭제하지 않으면 S3 버킷에 그대로 남아 있습니다.

DBR을 AWS CUR로 마이그레이션하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [the section called “DBR에서 AWS CUR로 마이그레이션”](#).

세부 결제 보고서를 비용 및 사용 보고서로 마이그레이션

세부 결제 보고서(DBR)와 AWS 비용 및 사용 보고서(AWS CUR)는 모두 요금에 대한 정보를 제공합니다. 그러나 DBR을 사용하는 경우 보고서를 비용 및 사용 보고서로 전환하는 것이 좋습니다.

주제

- [비용 및 사용 보고서\(AWS CUR\)의 이점 비교](#)
- [세부 결제 보고서와 비용 및 사용 보고서 간의 주요 차이점](#)
- [고급 요금 유형 보고](#)

비용 및 사용 보고서(AWS CUR)의 이점 비교

AWS CUR은 가장 포괄적인 정보 소스를 제공합니다. AWS CUR을 사용하여 개별 비용을 심층적으로 이해하고 더 자세히 분석할 수 있습니다. 이는 엔터프라이즈 규모에서 특히 유용합니다. AWS CUR은 복잡한 비용 관리 요구 사항이 있고 전용 쿼리 또는 분석 기반 시스템이 필요한 경우 유용합니다. AWS CUR은 분할 상환 비용을 포함하여 예약 인스턴스(RI)에 대한 세부 정보도 제공합니다.

포괄적인 예약 정보

예약 인스턴스(RI) 또는 예약은 1년 또는 3년의 서비스 사용 약정에 대한 대가로 온디맨드 사용량과 비교하여 할인된 시간당 요금을 제공합니다. 이 방법을 선택하면 결과적으로 상당한 비용을 절약할 수 있습니다. AWS CUR을 사용하여 예약 포트폴리오를 모니터링하고 관리할 수 있습니다. AWS CUR은 예

약 Amazon 리소스 번호(ARNs), 예약 수, 총 RIs와 같은 자세한 정보를 제공합니다. 특정 리소스의 예약 관련 할인을 추적하여 절약 금액에 대해 더 잘 이해할 수 있습니다.

세부 결제 보고서(DBR)는 이 메타데이터의 서브넷을 제공하지만, 필수 열을 전환하려면 작업이 필요합니다.

AWS CUR은 분할 상환된 예약 비용에 대한 정보와 같이 DBR에서 사용할 수 없는 추가 열을 제공합니다. 자세한 내용은 [the section called “분할 상환 예약 데이터 이해”](#) 단원을 참조하십시오.

온디맨드 요금 가용성

AWS CUR은 각 개별 사용 항목의 온디맨드 요금에 대한 정보를 제공합니다. 이 정보를 사용하여 온디맨드 요금에서 지불한 금액을 차감하여 절감액을 계산할 수 있습니다. 이는 퍼블릭 온디맨드 요금을 사용하여 비용을 할당하도록 선택할 수 있는 유연성도 제공합니다.

DBR에는 온디맨드 요금에 대한 정보가 포함되지 않으며 청구된 금액만 포함됩니다. 따라서 전체 절약 금액을 계산하거나 온디맨드 요금을 사용하여 비용을 할당하기가 어려울 수 있습니다.

세분화된 할인 분석

AWS CUR은 사용량 기반 할인에 대한 세분화된 보기에 액세스할 수 있습니다. 할인이 적용된 경우 AWS CUR을 사용하여 다음을 볼 수 있습니다.

- 할인 전 비용
- 할인된 금액
- 항목 수준에서 할인이 적용된 후 총 비용

DBR에는 세분화된 할인 분석이 포함되지 않습니다.

자동화된 대규모 데이터 수집

AWS CUR을 사용하면 자동 데이터 수집 프로세스를 트리거하도록 이벤트를 쉽게 구성하여 사내 시스템에서 결제 데이터를 새로 고치는 프로세스를 간소화할 수 있습니다. AWS CUR 데이터는 이전 달과 관련된 요금이 감지되면 자동으로 새로 고칠 수 있습니다.

또한 AWS CUR은 여러 파일로 생성되므로 데이터를 더 작은 조각으로 분할할 때 추가적인 이점을 얻을 수 있습니다. 따라서 여러 작업자가 사용하는 프로세스에 따라 데이터를 더 쉽게 수집할 수 있습니다. 또한 데이터 다운로드를 더 작은 단위로 다시 시도할 수 있습니다.

AWS CUR은 데이터를 빠르게 찾고 추출할 수 있는 방식으로 서식이 지정됩니다. 이 보고서는 데이터의 전체 구조에 대한 정보가 포함된 매니페스트 파일에서 모델링됩니다. 여기에는 보고서에 포함된 모

든 열의 목록이 포함됩니다. 이 정보를 사용하여 보고서를 확장하고 사용 가능해지면 사용량에 관한 새로운 정보를 포함할 수 있습니다.

교차 제품 통합

AWS CUR은 Amazon Redshift, Quick 및 Amazon Athena와 통합됩니다. AWS CUR을 사용하여 AWS 기반 비용 관리 솔루션을 구축할 수 있습니다. AWS CUR은 Parquet 형식의 데이터도 제공합니다. 이를 통해 자체 비용 및 사용량 보고 시스템을 구축할 수 있는 더 많은 옵션이 제공됩니다. 자세한 내용은 AWS Billing 사용 설명서의 [AWS Cost & Usage Report\(AWS CUR\) 매니페스트](#)를 참조하세요.

세부 결제 보고서와 비용 및 사용 보고서 간의 주요 차이점

CUR로 마이그레이션한 후 고려해야 AWS 할 DBR과 AWS CUR에는 몇 가지 차이점이 있습니다. 예를 들어 데이터를 시스템으로 수집하는 방법을 조정해야 할 수 있습니다.

파일 구조

세부 결제 보고서(DBR)는 단일 파일로 제공됩니다. 반면 AWS CUR은 통합된 파일 세트입니다. AWS CUR에서는 Amazon S3 버킷에서 다음 파일을 볼 수 있습니다.

- 모든 사용 항목이 포함된 데이터 파일 세트
- (해당되는 경우) 모든 할인이 포함된 별도의 데이터 파일
- 단일 보고서에 속한 모든 데이터 파일이 나열된 매니페스트 파일

열 구조

DBR에는 고정된 열 목록이 있으므로 유연성이 제한됩니다. AWS CUR에는 고정된 열 구조가 없으며 대신 필요에 따라 열을 자유롭게 추가하거나 제거할 수 있습니다. 새를 사용하기 시작하면 AWS 서비스 AWS CUR은 사례에 유용할 수 있는 새 데이터를 보고서에 포함하도록 동적으로 시작할 수 있습니다. 매니페스트 파일은 보고서에 있는 모든 열의 맵을 제공합니다.

DBR 및 AWS CUR에 해당하는 열 이름

DBR 열 이름	AWS CUR 열 이름
Invoiceld	bill/Invoiceld
PayerAccountld	bill/PayerAccountld
LinkedAccountld	lineItem/UsageAccountld

DBR 열 이름	AWS CUR 열 이름
ProductName	product/ProductName
SubscriptionId	reservation/subscriptionid
UsageType	lineItem/UsageType
연산	lineItem/Operation
AvailabilityZone	lineItem/AvailabilityZone
ReservedInstance	지원되지 않음
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	지원되지 않음
PricingplanId	지원되지 않음
RateID	pricing/RateId

Note

AWS CUR의 RecordId와 동등한 것은 없습니다. 그러나 identity/LineItemId, identity/TimeInterval 및 bill/BillType을 결합하여 이 정보를 수집할 수 있습니다.

AWS CUR을 통해 DBR RecordType 값 검색

DBR의 RecordType 값	AWS CUR을 통해 RecordType을 검색하는 구문	사용 사례:
LineItem	SELECT SUM(line_item_unblended_cost) FROM [CUR] WHERE line_item_line_item_type = 'Usage'	사용 항목은 사용 비용을 일회성 요금에서 분할합니다(예: 선결제 RI 지불).
InvoiceTotal	SELECT (bill_invoice_id), sum(line_item_unblended_cost) FROM [CUR] GROUP BY bill_invoice_id	인보이스 합계를 사용하여 인보이스와 비용 및 사용 보고서 간에 비용을 조정할 수 있습니다.
AccountTotal	SELECT line_item_usage_account_id, sum(line_item_unblended_cost) FROM [CUR] GROUP BY line_item_usage_account_id	계정 합계를 사용하여 지불 거절을 위해 멤버 계정과 관련된 비용을 격리할 수 있습니다.
StatementTotal	SELECT SUM(line_item_unblended_cost) FROM [CUR]	명세서 합계를 사용하여 결제 기간에 대한 비용을 이해할 수 있습니다.
Discount	SELECT SUM(line_item_unblended_cost) FROM [CUR] WHERE line_item_line_item_type = 'Discount'	할인 항목을 사용하여 모든 할인 관련 항목을 식별할 수 있습니다.
Rounding	아직 지원되지 않음	아직 지원되지 않음

고급 요금 유형 보고

환불

AWS CUR: `lineItem/LineItemDescription` = 'Refund' 문자열을 필터링하여 환불을 식별합니다.

DBR: 'Refund' 하위 문자열에 대해 `ItemDescription` 열을 확인하여 환불을 식별합니다.

Credits

AWS CUR: `lineItem/LineItemDescription` = 'Credit' 문자열을 필터링하여 크레딧을 식별합니다.

DBR: 'Credit' 하위 문자열에 대해 `ItemDescription` 열을 확인하여 크레딧을 식별합니다.

세금

AWS CUR: `lineItem/LineItemDescription` = 'Tax' 문자열을 필터링하여 세금을 식별합니다.

DBR: 'Tax' 하위 문자열에 대해 `ItemDescription` 열을 확인하여 세금을 식별합니다.

예약 관련 선불 요금 식별

AWS CUR: `"lineItem/LineItemType"` = 'Fee' 문자열을 필터링하여 예약 관련 선결제 비용을 식별합니다.

DBR: 'HeavyUsage' 하위 문자열의 `UsageType` 열 및 '`SubscriptionId`'가 null인지 여부를 검토하여 예약 관련 선결제 비용을 식별합니다.

예약 관련 월별 요금 식별

AWS CUR: `"lineItem/LineItemType"` = 'RIfee' 문자열을 필터링하여 예약 관련 월별 요금을 식별합니다.

DBR: 'HeavyUsage' 하위 문자열에 대해 `UsageType` 열을 확인하여 예약 관련 월별 요금을 식별합니다.

예약 인스턴스 이익을 얻은 인스턴스 식별

AWS CUR: `"lineItem/LineItemType"` = 'DiscountedUsage' 문자열을 필터링하여 예약 관련 선결제 요금을 식별합니다.

DBR: 'Y' 하위 문자열에 대해 `ReservedInstance` 열을 확인하여 예약 관련 선결제 요금을 식별합니다.

미사용 예약 비용 이해

AWS 비용 및 사용 보고서(AWS CUR)를 사용하여 미사용 RI 비용을 이해할 수 있습니다. 다음 네 가지 시나리오는 방법을 보여 줍니다.

Note

다음 표에서는 명확성을 위해 AWS CUR 및 DBR/DBR-RT의 열과 행이 변환됩니다. 첫 번째 열의 값은 보고서의 헤더를 나타냅니다.

시나리오 1: RI 사용량은 100%입니다.

RI 요금 항목에 0 USD 미사용 비용 및 0 사용 시간이 있습니다.

DBR/DBR-RT를 사용하면 RI 요금 항목에 대한 UsageQuantity 및 UnblendedCosts 필드를 참조하여 미사용 RI 사용량 및 비용을 이해할 수 있습니다. RI 요금 항목은 ItemDescription 필드에 '구입 시간' 정보가 있는지 여부로 식별할 수 있습니다. 표 1은 DBR 및 DBR-RT 보고서에서 미사용 RI 비용을 관리하는 데 사용되는 열 및 정보를 보여줍니다.

표 1 – 2019년 6월 17일 이전 DBR 및 DBR-RT에서 100% RI 사용량에 대한 미사용 RI 비용

ProductName	- Amazon Elastic Compute Cloud	- Amazon Elastic Compute Cloud
UsageType	HeavyUsage:c3.8xlarge	HeavyUsage:c3.8xlarge
작업	RunInstances	RunInstances
가용 영역	us-east-1a	us-east-1a
예약 인스턴스	Y	Y
ItemDescription	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge (744 hours	USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge

	purchased, 744 hours used)	
사용량	0	744
일반 요금	0.1	0.1
일반 비용	0	74.4

CUR을 사용하면 RI 요금 항목에 대한 AWS 'reservation/UnusedQuantity' 및 'reservation/UnusedRecurringFee' 필드를 참조하여 미사용 RI 사용량 및 비용을 이해할 수 있습니다. 아래 표 4는 AWS CUR에서 미사용 RI 비용을 관리하는 데 사용되는 현재 열과 정보를 보여줍니다.

표 2 - AWS CUR에서 100% RI 사용량에 대한 미사용 RI 비용

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/UnusedRecurring Fee	0	
reservation/UnusedAmortized UpfrontFeeForBillingPeriod	0	
reservation/RecurringFeeFor Usage		74.4
reservation/AmortizedUpfrontCostForUsage		5
reservation/EffectiveCost		79.4

DBR/DBR-RT에서 지원하는 현재 기능을 일치시키는 것 외에도 AWS CUR에는 다음과 같은 이점이 있습니다.

- AWS CUR을 사용하면 기본 요금과 선결제 요금을 모두 포함하는 DiscountedUsage 품목의 EffectiveCost와 관련된 정보에 액세스할 수 있습니다. DBR은 기본 요금만 보고합니다.
- AWS CUR에서 UsageType 필드는 DiscountedUsage 행 항목에 대해 변환되지 않는 반면 DBR은 정보를 RI 요금 행 항목 정보로 대체합니다. 이는 사용자가 AWS CUR의 행 항목을 ReservationARN별로 그룹화하여 어떤 RI가 어떤 사용량을 할인했는지 이해할 수 있기 때문입니다.
- AWS CUR에서 LineItemDescription 필드는 RI 요금 항목에 대해 변환되지 않습니다. DBR은 구입한 시간과 사용된 시간을 추가합니다.

시나리오 2: 부분 RI 사용량

RI 요금 항목에 미사용 비용 및 사용이 있습니다.

DBR/DBR-RT를 활용하면 RI 요금 항목에 대한 UsageQuantity 및 UnblendedCosts 필드를 참조하여 미사용 RI 사용 및 비용을 이해할 수 있습니다. 표 3은 DBR 및 DBR-RT 보고서에서 미사용 RI 비용을 관리하는 데 사용되는 열 및 정보를 보여줍니다.

표 3 – 2019년 6월 17일 이전 DBR 및 DBR-RT에서 부분 RI 사용량에 대한 미사용 RI 비용

ProductName	- Amazon Elastic Compute Cloud	- Amazon Elastic Compute Cloud
-------------	--------------------------------	--------------------------------

UsageType	HeavyUsage:c3.8xlarge	HeavyUsage:c3.8xlarge
작업	RunInstances	RunInstances
가용 영역	us-east-1a	us-east-1a
예약 인스턴스	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
사용량	100	644
일반 요금	0.1	0.1
일반 비용	10	64.4

CUR을 사용하면 RI 요금 항목에 대해 AWS 'reservation/UnusedQuantity' 및 'reservation/UnusedRecurringFee' 필드를 참조하여 미사용 RI 사용량 및 비용을 이해할 수 있습니다. 표 4는 AWS CUR에서 미사용 RI 비용을 관리하는 데 사용되는 현재 열과 정보를 보여줍니다.

표 4 - AWS CUR에서 부분 RI 사용량에 대한 미사용 RI 비용

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	USD 0.00 hourly fee per Linux/UNIX

	X (Amazon VPC), c3:8xlarge	X (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-RT에서 지원하는 현재 기능을 일치시키는 것 외에도 AWS CUR에는 다음과 같은 이점이 있습니다.

- AWS CUR에는 RI 요금 항목에 대한 UnusedQuantity와 미사용 시간으로 UsageQuantity 열을 오버로드하는 DBR/DBR-RT를 나타내는 별도의 열이 있습니다.

시나리오 3: 용량 예약

DBR/DBR-RT에서는 RI 요금 항목에 UsageQuantity 및 UnblendedCost 필드의 미사용량이 이미 포함되기 때문에 RI에 포함되는 경우 용량 예약 관련 UnusedBox 및 UnusedDed 사용 유형 항목이 필터링

됩니다. 표 5는 DBR 및 DBR-RT 보고서에서 미사용 RI 비용을 관리하는 데 활용되는 열 및 정보를 보여줍니다.

표 5 – 2019년 6월 17일 이전 DBR 및 DBR-RT에서 용량 예약 시나리오에 대한 미사용 RI 비용

ProductName	- Amazon Elastic Compute Cloud	- Amazon Elastic Compute Cloud
UsageType	HeavyUsage:c3.8xlarge	HeavyUsage:c3.8xlarge
작업	RunInstances	RunInstances
가용 영역	us-east-1a	us-east-1a
예약 인스턴스	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
사용량	10	734
일반 요금	0.1	0.1
일반 비용	1	73.4

AWS CUR은 이러한 행 항목을 DiscountedUsage로 표시합니다. 표 6은 AWS CUR에서 미사용 RI 비용을 관리하는 데 사용되는 현재 열과 정보를 보여줍니다.

표 6 - CUR의 용량 예약 시나리오에 대한 미사용 AWS RI 비용

lineitem/ProductCode	Amazon EC2	Amazon EC2	Amazon EC2
----------------------	------------	------------	------------

UsageType	HeavyUsage: c3.8xlarge	USW2-Rese rvation: c3.8xlarge	USW2-BoxUsage: c3.8xlarge
lineitem/LineItemType	RI Fee	Usage	DiscountedUsage
lineitem/LineItemDescription	USD 0.10 hourly fee per Linux/ UNIX (Amazon VPC), c3:8xlarg e	USD 0.00 per Reservation Linux/UNIX (Amazon VPC), c3:8xlarge Instance Hour	USD 0.00 hourly fee per Linux/ UNIX (Amazon VPC), c3:8xlarg e
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

시나리오 4: 유연한 크기의 예약

DBR/DBR-RT를 활용하면 RI 요금 항목에 대한 UsageQuantity 및 UnblendedCosts 필드를 참조하여 미사용 RI 사용 및 비용을 이해할 수 있습니다. RI 요금 항목은 ItemDescription 필드에 '구입 시간' 정보가 있는지 여부로 식별할 수 있습니다. 표 9는 DBR 및 DBR-RT 보고서에서 미사용 RI 비용을 관리하는 데 활용되는 열 및 정보를 보여줍니다.

표 7 – 2019년 6월 17일 이전 DBR 및 DBR-RT에서 유연한 크기의 RI 시나리오에 대한 미사용 RI 비용

ProductName	- Amazon Elastic Compute Cloud	- Amazon Elastic Compute Cloud
UsageType	HeavyUsage:c3.8xlarge	HeavyUsage:c3.8xlarge
작업	RunInstances	RunInstances
가용 영역	us-east-1a	us-east-1a
예약 인스턴스	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
사용량	100	644
일반 요금	0.1	0.1
일반 비용	10	64.4

CUR을 사용하면 RI 요금 항목에 대해 AWS 'reservation/UnusedQuantity' 및 'reservation/UnusedRecurringFee' 필드를 참조하여 미사용 RI 사용량 및 비용을 이해할 수 있습니다. 표 8은 AWS CUR에서 미사용 RI 비용을 관리하는 데 사용되는 현재 열과 정보를 보여줍니다.

표 8 - AWS CUR에서 유연한 크기의 RI 시나리오에 대한 미사용 RI 비용

lineitem/ProductCode	Amazon EC2	Amazon EC2
UsageType	HeavyUsage:c3.8xlarge	USW2-BoxUsage:c3.8xlarge
lineitem/LineltemType	RI Fee	DiscountedUsage
lineitem/LineltemDescription	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/UnusedRecurring Fee	70.37	
reservation/UnusedAmortized UpfrontFeeForBillingPeriod	5.5	
reservation/RecurringFeeFor Usage		4.03
reservation/AmortizedUpfrontCostForUsage		0.5
reservation/EffectiveCost		4.53

DBR/DBR-RT에서 지원하는 현재 기능을 일치시키는 것 외에도 AWS CUR에는 다음과 같은 이점이 있습니다.

- AWS CUR에는 NormalizedUsageAmount 및 수량이 있습니다. DBR/DBR-RT에는 이 정보를 나타내는 열이 없습니다.
- AWS CUR UsageType 및 작업은 DiscountedUsage 항목에 대해 변환되지 않습니다. DBR/DBR-RT에서는 이러한 값이 RI 요금 항목으로 대체됩니다.
- AWS CUR LineItemDescription은 DiscountedUsage 행 항목에 대해 변환되지 않습니다. DBR/DBR-RT에서는 RI 요금 항목 설명으로 대체하고 DiscountedUsage 항목 Usage Type을 문자열의 끝에 추가합니다. 예: “USD 0.10 hourly fee per Linux/UNIX (Amazon VPC), c3:8xlarge; UsageType: BoxUsage:c3.large”

월별 보고서 다운로드

Billing and Cost Management 콘솔의 청구서 페이지에서 예상 AWS 요금에 대한 월별 보고서를 다운로드할 수 있습니다.

에서 통합 결제 기능을 사용한다고 가정해 보겠습니다 AWS Organizations. 그러면 이 보고서는 관리 계정에서만 사용할 수 있으며 모든 멤버 계정의 활동이 포함됩니다. 멤버 계정 소유자는 관리 계정에서만 월별 보고서를 구할 수 있습니다. 자세한 정보는 AWS Billing 사용 설명서의 [조직의 통합 결제](#)를 참조하세요.

보고서에는 계정에서 사용하는 AWS 제품, 사용 유형 및 작업의 각 고유 조합에 대한 항목이 포함되어 있습니다. 예상 보고서는 하루에도 여러 차례 업데이트될 수 있습니다. 명세서 기간을 선택하여 이전 달의 보고서를 가져올 수 있습니다. 월별 보고서를 신청한 달의 보고서부터 시작합니다. 가입하기 전의 보고서는 사용할 수 없습니다.

월별 비용 할당 보고서 다운로드

Important

나중에 월별 비용 할당 보고서 기능은 사용할 수 없습니다. 대신 AWS 비용 및 사용 보고서를 사용하는 것이 좋습니다.

AWS 사용량의 비즈니스 차원을 설명할 수 있는 AWS 리소스에 대한 사용자 지정 비용 할당 태그 세트를 생성할 수 있습니다. 이러한 태그 세트를 사용하여 AWS 비용을 구성하고 추적할 수 있습니다. 많은 가 특성 세트에 태그 지정을 AWS 서비스 노출합니다. 콘솔, API 또는 AWS 명령줄 인터페이스(CLI)을 사용하여 이러한 서비스 내에서 태그를 생성합니다. 자세한 내용은 AWS Billing 사용 설명서의 [비용 할당 태그 사용](#)을 참조하세요.

태그를 생성한 후 월별 비용 할당 보고서를 얻을 수 있습니다. 이는 기본적으로 비용 할당 태그 세트를 포함하는 월별 보고서입니다.

AWS 사용 보고서 다운로드

Important

2023년 9월 15일에 AWS 사용 보고서는 2019년 3월 1일 이전의 사용 데이터에 대한 액세스를 더 이상 제공하지 않습니다. 이러한 사용 데이터에 액세스하려면 2023년 9월 15일 이전에 사용량 기록을 다운로드하여 로컬에 저장합니다. AWS 사용 보고서 기능은 나중에 사용할 수 없습니다. 대신 AWS 비용 및 사용 보고서를 사용하는 것이 좋습니다.

사용 보고서를 XML 또는 CSV 형식으로 다운로드할 수 있습니다. 보고서는 사용 유형, 작업 및 기간을 기준으로 단일 서비스를 다룹니다. 데이터를 집계하는 방법을 선택할 수도 있습니다.

사용 보고서 다운로드

1. <https://console.aws.amazon.com/costmanagement/>에서 과금 정보 및 비용 관리 콘솔을 엽니다.
2. 탐색 창에서 레거시 페이지를 선택하고 비용 및 사용 보고서를 선택합니다.
3. AWS 사용 보고서 섹션에서 사용 보고서 생성을 선택합니다.
4. 사용 보고서 다운로드 페이지의 서비스에서 사용량을 보려는 서비스를 선택합니다.
5. 사용 유형을 선택합니다.
6. 작업을 선택합니다.
7. 보고서 기간을 선택합니다. 사용자 지정 날짜 범위를 선택하는 경우 보고서의 날짜 범위를 수동으로 지정해야 합니다.
8. 보고서 세부 수준에는 시간별, 일별 또는 월별을 선택합니다.
9. 다운로드를 선택한 다음 XML 보고서 또는 CSV 보고서를 선택합니다.

Note

큰 보고서를 다운로드하면 보고서 내용이 잘릴 수 있습니다. 다운로드한 파일의 마지막 행에서 경고 또는 오류 메시지를 확인합니다. 보고서가 잘린 경우 기간을 짧게 선택하여 더 작은 보고서를 다운로드합니다. 또 다른 옵션은 보고서 세부 수준을 시간별에서 일별 또는 월별로 줄이는 것입니다.

비용 및 사용 보고서 문제 해결

다음 항목을 사용하여 비용 및 사용 보고서와 관련된 문제를 해결할 수 있습니다.

주제

- [Amazon S3 버킷에 보고서 파일이 없음](#)
- [보고서 데이터 파티션 중 하나가 비어 있음](#)
- [내 비용 및 사용 보고서 데이터가 다른 결제 및 비용 관리 기능의 데이터와 일치하지 않음](#)
- [보고서 설정을 변경했기 때문에 데이터를 채우고 싶음](#)
- [Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있음](#)
- [보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없음](#)
- [Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않음](#)
- [보고서에 포함된 열이 전월과 달라짐](#)
- [보고서의 열이 변경되어 보고서에 기반한 쿼리나 테이블이 작동하지 않음](#)
- [보고서를 쿼리하는 데 도움이 필요함](#)
- [Amazon EC2 전용 호스트의 결제 데이터를 찾을 수 없음](#)
- [Amazon EC2 탄력적 IP 주소의 결제 데이터를 이해할 수 없음](#)
- [통합 결제를 사용하는데 일반 요금 및 혼합 요금이나 비용의 차이를 이해할 수 없음](#)
- [보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0임](#)
- [보고서에서 전체 선결제 예약형 인스턴스가 분할 상환되는 방식을 이해할 수 없음](#)

Amazon S3 버킷에 보고서 파일이 없음

Amazon S3 버킷 정책이 파일을 버킷에 넣을 수 있는 billingreports.amazonaws.com 서비스 권한을 부여하는지 확인합니다. 필수 버킷 정책에 대한 자세한 정보는 [비용 및 사용 보고서에 대한 Amazon S3 버킷 설정](#) 섹션을 참조하세요.

보고서 데이터 파티션 중 하나가 비어 있음

보고서가 대부분의 애플리케이션에서 처리할 수 있는 것보다 크면는 보고서를 여러 파일로 AWS 분할합니다. 보고서 업데이트는 이전 보고서 버전보다 개별 파일 파티션 수가 적을 수 있습니다.

보고서의 매니페스트 파일을 검토하여 수집할 필요가 없는 빈 파일을 찾습니다.

내 비용 및 사용 보고서 데이터가 다른 결제 및 비용 관리 기능의 데이터와 일치하지 않음

기타 결제 및 비용 관리 기능(Cost Explorer, 세부 결제 보고서, 결제 및 비용 관리 콘솔)에서는 다음과 같은 이유로 비용이 다르게 표시될 수 있습니다.

- 결제 기능은 다양한 방식으로 비용 데이터를 반올림합니다.
- 결제 기능마다 데이터 새로 고침 설정이 다를 수 있습니다. 예를 들어 비용 및 사용 보고서에서 청구가 확정된 후 적용된 환불, 크레딧 또는 지원 요금으로 이전 마감된 청구를 자동으로 새로 고칠지 여부를 선택할 수 있습니다. Cost Explorer는 동일한 항목을 자동으로 반영합니다. 이 시나리오에서 비용 및 사용 보고서에서 자동 새로 고침을 활성화하지 않으면 비용 및 사용 보고서 데이터가 Cost Explorer 데이터와 일치하지 않습니다.
- 결제 기능은 요금을 다양하게 그룹화할 수 있습니다. 예를 들어 결제 및 비용 관리 콘솔의 청구서 페이지에는 데이터 전송 요금이 AWS 서비스 요금 내에서 별도의 데이터 전송 그룹으로 표시됩니다. 한편, 비용 및 사용 보고서와 Cost Explorer에는 데이터 전송 요금이 각 서비스의 사용 유형으로 표시됩니다.

이러한 이유를 검토한 후에도 비용 및 사용 보고서와 다른 결제 및 비용 관리 기능 간에 여전히 불일치가 있다고 생각되면 지원 사례를 열어 비용 데이터 검토를 요청합니다. 지원 사례에는 보고서 이름과 검토할 결제 기간을 입력해야 합니다. 사례 열기에 대한 자세한 내용은 [내보내기 및 보고서 관련 도움말](#) 섹션을 참조하세요.

보고서 설정을 변경했기 때문에 데이터를 채우고 싶음

지원 사례를 열어 비용 데이터 채우기를 요청합니다. 지원 사례에는 보고서 이름과 채울 결제 기간을 입력해야 합니다. 사례 열기에 대한 자세한 내용은 [내보내기 및 보고서 관련 도움말](#) 섹션을 참조하세요.

단, 다음과 같은 시나리오에서는 비용 데이터를 다시 채울 수 없습니다.

- 계정을 만든 날짜 이전의 비용 데이터는 다시 채울 수 없습니다.
- AWS Organizations 를 사용하고 관리 계정으로 지정된 계정과 같이 조직의 구조가 변경된 경우 이전 조직 구조로 데이터를 채울 수 없습니다.
- AWS Organizations 를 사용하고 조직을 변경하는 경우 현재 조직에 가입하기 전에에서 데이터 채우기를 가져올 수 없습니다.

Amazon S3의 보고서 파일 폴더가 이름이 지정되지 않은 폴더에 있음

보고서의 보고서 경로 접두사에 / 문자가 있으면 Amazon S3 버킷에 이름 없는 폴더가 생성됩니다. 다음 보고서 업데이트에서 이름이 지정되지 않은 폴더를 제거하려면 보고서 설정을 편집하고 보고서 경로 접두사에서 / 문자를 제거합니다. 지침은 [비용 및 사용 보고서 구성 편집](#) 섹션을 참조하세요.

보고서에 리소스 ID를 포함하는 옵션을 선택할 수 없음

보고서를 생성할 때 리소스 ID 포함 옵션을 선택할 수 있습니다. 보고서 버전 관리를 기존 보고서 덮어쓰기로 설정한 상태에서 보고서를 생성하는 경우 보고서를 생성한 후에는 리소스 ID 포함 선택을 수정할 수 없습니다. 리소스 ID를 포함하려면 새 보고서를 만들고 리소스 ID 포함 옵션을 선택해야 합니다.

Amazon Athena에 대한 비용 및 사용 보고서 쿼리가 Amazon Redshift에서 작동하지 않거나 Amazon Redshift 쿼리가 Amazon Athena에서 작동하지 않음

Amazon Athena와 Amazon Redshift 데이터베이스는 비용 및 사용 보고서 열의 형식을 다르게 지정합니다. Amazon Athena는 열 이름(line_item_normalized_usage_amount)의 단어 사이에 밑줄을 추가합니다. Amazon Redshift는 열 유형과 속성(linitem_normalizeduseamount) 사이에 밑줄을 추가합니다. Amazon Athena 또는 Amazon Redshift의 열 이름 형식과 일치하도록 쿼리를 수정해야 합니다.

보고서에 포함된 열이 전월과 달라짐

가 보고서에 AWS 포함하는 열은 AWS 사용량에 따라 다릅니다. 모든 보고서에는 identity/, bill/, lineltem/ 접두사가 있는 열이 포함됩니다.

- identity/LineltemId
- identity/TimeInterval
- bill/InvoiceId
- bill/BillingEntity
- bill/BillType
- bill/PayerAccountId
- bill/BillingPeriodStartDate
- bill/BillingPeriodEndDate
- lineltem/UsageAccountId
- lineltem/LineltemType

- 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

다른 모든 열은 월별 AWS 사용량이 해당 열을 채우는 데이터를 생성하는 경우에만 포함됩니다.

예를 들어, 보고서에는 해당 월에 절감형 플랜을 사용한 경우에만 savingsPlan/ 열이 포함됩니다.

보고서의 열이 변경되어 보고서에 기반한 쿼리나 테이블이 작동하지 않음

가 보고서에 AWS 포함하는 열은 해당 월의 AWS 사용량에 따라 달라집니다. 보고서에 포함된 열은 변경될 수 있으므로 보고서를 기반으로 하는 사용자 지정 쿼리나 표에서는 열 번호 대신 열 이름을 참조하는 것이 좋습니다.

보고서를 쿼리하는 데 도움이 필요함

비용 및 사용 보고서 쿼리에 대한 자세한 내용은 AWS Well-Architected Labs 웹 사이트의 [CUR 쿼리 라이브러리 도움말](#)을 참조하세요.

Amazon EC2 전용 호스트의 결제 데이터를 찾을 수 없음

ResourceID 옆에서 인스턴스 ID가 아닌 전용 호스트 ID를 찾습니다. 전용 호스트는 전용 호스트 실행 시간을 기준으로 측정되므로 보고서에는 호스트 ID와 관련된 측정 시간별로 전용 호스트 사용량이 표시됩니다.

Amazon EC2 탄력적 IP 주소의 결제 데이터를 이해할 수 없음

Amazon EC2 탄력적 IP 주소는 집계되어 측정됩니다. 즉, 보고서의 각 항목은 개별 탄력적 IP 주소와 일치하지 않습니다. 각 항목은 청구 가능한 총 시간을 나타냅니다. 실행 중인 인스턴스와 연결되어 있는 탄력적 IP 주소는 한 개까지 무료로 사용할 수 있습니다. 인스턴스에 할당한 추가 탄력적 IP 주소 하나당 시간당 요금이 비례 할당됩니다. 또한 할당되지 않은 탄력적 IP 주소에 대해 시간당 요금이 AWS 부과됩니다.

통합 결제를 사용하는데 일반 요금 및 혼합 요금이나 비용의 차이를 이해할 수 없음

에 대한 통합 결제 AWS Organizations를 사용하면 혼합되지 않은 혼합 요금 또는 비용을 통해 독립 실행형 계정과 조직의 연결된 계정에 대한 계정 사용 비용을 이해하는 데 도움이 될 수 있습니다. 일부 서비스는 사용량 증가에 따라 단가를 낮출 수 있는 비용 계층을 제공합니다. 는 조직의 서비스에 대한 모든 사용량을 AWS 집계하므로 개별 계정은 조직의 월별 사용량에 사용량이 집계될 때 더 빨리 더 저렴한 티어에 액세스할 수 있습니다.

일반 요금은 개별 계정의 서비스 사용과 관련된 요금입니다. 항목의 경우 일반 비용은 사용량에 일반 요금을 곱한 값입니다. 독립 실행형 계정의 경우 일반 비용은 계정 사용 비용입니다. 혼합 요금은 조직의 총 사용량과 관련된 요금으로, 계정 전체의 평균을 구합니다. 항목의 경우 혼합 비용은 사용량에 혼합 요금을 곱한 값입니다. 혼합 비용은 조직의 연결 계정으로 계정을 사용하는 데 따른 비용입니다.

일반 및 혼합 비용 계산에 대한 자세한 정보 및 예제는 AWS Billing 사용 설명서의 [통합 청구서 이해](#)를 참조하세요.

보고서 일부 항목의 혼합 요금 또는 혼합 비용이 0임

예약형 인스턴스 할인이 적용되는 Amazon EC2 항목의 혼합 요금은 0입니다. 이러한 항목의 경우 LineltemType은 사용량 할인입니다.

혼합 비용은 사용량에 혼합 요금을 곱한 값입니다. 혼합 요금 또는 사용량 값이 0이면 혼합 비용도 0입니다.

보고서에서 전체 선결제 예약형 인스턴스가 분할 상환되는 방식을 이해할 수 없음

모든 선결제 예약형 인스턴스는 전액 선결제로 청구되므로 분할 상환된 비용은 선결제 금액을 관련 기간(1년 또는 3년)으로 나누어 보고서에 반영됩니다.

reservation/AmortizedUpfrontCostForUsage 및 reservation/EffectiveCost는 모든 선결제 예약형 인스턴스의 요금과 동일합니다. 이는 두 항목 모두 예약형 인스턴스에 대한 선결제 금액을 계약 기간의 총 시간에 걸쳐 균등하게 분배하기 때문입니다.

RIFee가 0.00 USD이긴 하지만 보고서에는 모든 선결제 예약형 인스턴스에 대한 RIFee 항목이 입력되어 있을 것으로 예상됩니다. 이러한 항목은 해당 월의 시간당 기본 요금을 나타내며 다른 열에는 추가 사용 데이터가 있습니다. 모든 예약형 인스턴스는 RIFee 항목을 생성합니다.

AWS 비용 및 사용 보고서의 보안

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 보안에 가장 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

AWS 비용 및 사용 보고서는 AWS 결제 및 비용 관리 콘솔의 기능입니다. 보안 고려 사항에 대한 자세한 정보는 AWS Billing 사용 설명서의 [AWS 결제 및 비용 관리의 보안](#)을 참조하세요.

AWS CUR을 사용하기 위한 액세스 제어 및 IAM 권한에 대한 자세한 내용은 [액세스 권한 관리 개요를 참조하세요](#).

할당량 및 제한

다음 표에서는 AWS Cost & Usage Report(AWS CUR)의 현재 할당량 및 제한에 대해 설명합니다.

비용 및 사용 보고서

비용 및 사용 보고서 수	계정당 10
비율	비용 및 사용 보고서는 무료이지만 표준 Amazon S3 요금이 적용됩니다.
프리 티어 AWS Cost & Usage Report(AWS CUR) 수	10

내보내기 및 보고서 관련 도움말

AWS 결제 및 비용 관리, AWS Data Exports, AWS비용 및 사용량 보고서 질문에 관해 도움을 받을 수 있는 다양한 리소스가 준비되어 있습니다.

- [AWS 지식 센터](#): 내보내기 및 보고서 관련 질문에 대한 답변을 가장 빠르게 찾을 수 있는 방법입니다. 여기에서 시작하는 것이 좋습니다.
- 계정 및 결제 지원: AWS 계정 소유자인 경우 계정 및 결제 지원에 무료로 액세스할 수 있습니다. 맞춤형 기술 지원에만 지원 플랜이 필요합니다. 자세한 내용은 [지원](#)에서 알아보세요.
- 지원 사례 열기: AWS Support에 연락하여 문의에 대한 지원 사례를 개설할 수 있습니다. 이는 AWS Support와 커뮤니케이션하는 가장 직접적인 방법입니다. 지원에서는 담당자에게 연락할 수 있는 직통 전화번호를 게시하지 않고 다음 절차를 통해 전화를 겁니다.

Note


지원 사례를 열고 관련 주제: 계정 및 결제 지원을 지정하려면 루트 계정 소유자로 AWS에 로그인하거나 지원 사례를 열 수 있는 IAM 권한이 있어야 합니다. 자세한 내용은 지원 사용 설명서의 [지원 시작하기](#)를 참조하세요.

AWS 계정을 폐쇄한 경우에도 지원에 로그인하여 과거 청구서를 볼 수 있습니다.

AWS Support에 문의하려면 다음과 같이 하세요.

1. 로그인하고 [지원센터](#)로 이동합니다.
2. 사례 생성을 선택합니다.
3. 사례 생성 페이지에서 계정 및 결제 지원을 선택하고 양식의 필수 필드를 작성합니다.
4. 다음 단계: 추가 정보를 선택합니다
5. 추가 정보(Additional information) 페이지의 제목(Subject)에 해당 문제에 대한 제목을 입력합니다.
6. 설명에 질문 또는 문제를 자세히 설명합니다.
7. (선택 사항) 파일 첨부(Attach files)를 선택하여 오류 로그 또는 스크린샷과 같은 관련 파일을 사례에 추가합니다. 최대 3개의 파일을 첨부할 수 있습니다. 각 파일의 크기는 최대 5MB까지입니다.
8. 다음 단계: 지금 해결하거나 문의하기를 선택합니다.
9. 문의처 페이지에서 선호하는 언어를 선택합니다.
10. 선호하는 연락 방법을 선택합니다. 다음 옵션 중 하나를 선택할 수 있습니다.

- 웹(Web): 지원 센터에서 답변을 받습니다.
- 전화: 지원 담당자로부터 전화를 받습니다.

 Note

결제 문의는 인스턴트 메시징으로는 지원되지 않습니다.

11. 사례 세부 정보를 검토한 다음 제출을 선택합니다. 사례 ID 번호와 요약이 표시됩니다.

문서 이력

다음 표에서는 이 릴리스에 대한 설명서를 설명합니다 AWS Data Exports.

변경 사항	설명	날짜
데이터 내보내기에서 용량 예약 지원	이제 데이터 내보내는 지정된 품목에 적용되는 용량 예약에 대한 정보를 지원합니다.	2025년 11월 13일
Amazon EKS의 분할 비용 할당 데이터에서 Kubernetes 레이블 지원	분할 비용 할당 데이터가 이제 Amazon EKS 클러스터의 비용 할당 태그로 Kubernetes 레이블을 지원합니다.	2025년 10월 27일
Amazon EKS 분할 비용 할당 데이터에 GPU 지원 추가	분할 비용 할당 데이터에 GPU 리소스 예약 데이터가 포함되어 고객이 GPU 가속 워크로드 비용을 추적할 수 있습니다.	2025년 9월 1일
비용 최적화 기회에서 계정 이름 지원 추가	계정 이름을 사용하여 비용 최적화 권장 사항을 쉽게 확인, 필터링, 통합 및 우선순위 지정할 수 있습니다.	2025년 7월 23일
위치 기반 배출량 지원 추가	기존 시장 기반 방법(MBM)과 함께 위치 기반 방법(LBM)을 사용하여 계산된 탄소 배출량을 데이터 내보내기에서 확인할 수 있습니다.	2025년 6월 24일
탄소 배출량 데이터 내보내기 추가	Data Exports에서 탄소 배출량 데이터 내보내기를 생성할 수 있습니다.	2025년 4월 23일
GA에서 FinOps 파운데이션 오픈 소스(AWS 열이 있는	일반 공급(GA) 단계의 FinOps Open Cost and Usage Specification(FOCUS)을 사용	2024년 11월 25일

FOCUS 1.0)를 사용하여 내보내기 추가	하여 데이터 내보내기에서 비용 및 사용 내보내기를 생성할 수 있습니다.	
비용 할당 데이터 분할 및 Amazon CloudWatch Container Insights	Amazon EKS에 대한 비용 할당 데이터 분할 기능이 이제 Amazon CloudWatch Container Insights의 지표를 지원합니다.	2024년 11월 14일
FinOps 파운데이션 오픈 소스를 사용하여 내보내기 추가 (AWS 열이 있는 FOCUS 1.0 - 미리 보기)	FinOps Open Cost and Usage Specification(FOCUS)을 사용하여 데이터 내보내기에서 비용 및 사용량 내보내기를 생성할 수 있습니다.	2024년 6월 20일
비용 최적화 권장 사항 내보내기 추가(Cost Optimization Hub에서)	데이터 내보내기에서 비용 최적화 권장 사항의 내보내기를 생성할 수 있습니다(Cost Optimization Hub에서).	2024년 6월 20일
비용 할당 데이터 분할 및 Amazon Managed Service for Prometheus	Amazon EKS의 비용 할당 데이터 분할 기능이 이제 Amazon Managed Service for Prometheus의 지표를 지원합니다.	2024년 6월 10일
Amazon EKS 및 비용 할당 데이터 분할 통합	비용 할당 데이터 분할 기능을 확장하여 Amazon EKS와 통합합니다.	2024년 4월 16일

AWS Data Exports 시작됨

AWS Data Exports 를 사용하면 열 선택 및 행 필터링에 SQL 을 사용하여 비용 및 사용 보고서(CUR) 2.0의 내보내기를 생성할 수 있습니다. 이것은 AWS 로부터 가장 상세한 비용 및 사용량 데이터를 받을 수 있는 새로운 방법이자 권장되는 방법입니다.

2023년 11월 26일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.