



API Reference

AWS Marketplace



AWS Marketplace: API Reference

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

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
About the APIs	2
Using the APIs	2
Using the Catalog API	2
Using the Agreement API	18
Using the Metering API	19
Using the Entitlement API	20
Using the Deployment API	20
Using the Discovery API	21
Supported Regions	21
Regions for the Catalog API	21
Regions for the Agreement API	22
Regions for the Metering API	22
Regions for the Entitlement API	24
Regions for the Deployment API	24
Permissions	24
Access for the Catalog API	24
Access for the Agreement API	42
Access for the Metering API	42
Access for the Entitlement API	42
Access for the Deployment API	43
Access for the Discovery API	43
Access control for the AWS Marketplace Reporting API	44
Quotas	46
Quotas for the Catalog API	46
Quotas for the Agreement API	48
Quotas for the Deployment API	48
Logging	50
Logging the Catalog API	50
Logging the Agreement API	52
Logging the Deployment API	59
Logging the Discovery API	62
Notifications	65
Notifications for the Catalog API	65

Using the API as a seller	66
Work with seller products	66
Create a product	68
Update product details	71
Add pricing dimensions	79
Update pricing dimensions	86
Restrict pricing dimensions	91
Update targeting configuration	95
Update product visibility	98
Publish a product	103
Find your product ID	105
Change set status and errors	111
Work with single AMI products	119
Work with EC2 Image Builder component products	148
Work with container-based products	155
Work with SaaS products	194
Work with offers	209
Work with private offers	210
Work with resale authorizations	274
Work with channel partner private offers	323
Work with renewals	374
Using the API as a buyer	378
Work with the catalog to discover products	378
Work with the private marketplace	378
Creating a private marketplace	380
Changing the branding of a private marketplace experience	384
Enabling or disabling a private marketplace experience	386
Enabling or disabling user requests	387
Getting a list of products in a private marketplace experience	388
Adding or removing products from a private marketplace	391
Finding products	392
Working with private marketplaces for AWS Organizations	393
Associating principals to experiences	395
Archiving and reactivating a private marketplace experience	397
Errors in the private marketplace API	398
Entity types defined by private marketplace	401

Using the API to share resources	403
Prerequisites to share AWS Marketplace entities	403
Share an AWS Marketplace entity	403
Differences between sharing an entity through AWS RAM and the AWS Marketplace	
Catalog API	405
Attach read-only policy to your resource	406
Attach read and write resource policy to your resource	407
View resource policy set on your resource	408
Delete resource policy on your resource	409
View all resources owned by you and shared with you	409
Working with AWS SDKs	411
Code examples	413
AWS Marketplace Catalog API	417
AMI products	421
Channel partner offers	479
Container products	520
Entities	534
Offers	545
Products	682
Resale authorization	694
SaaS products	791
Utilities	848
AWS Marketplace Agreement API	855
Agreements	857
Actions	971
AWS Marketplace Catalog API	972
BatchDescribeEntities	973
CancelChangeSet	995
DeleteResourcePolicy	999
DescribeChangeSet	1002
DescribeEntity	1008
GetResourcePolicy	1013
ListChangeSets	1016
ListEntities	1021
ListTagsForResource	1028
PutResourcePolicy	1032

StartChangeSet	1035
TagResource	1042
UntagResource	1045
AWS Marketplace Agreement API	1047
DescribeAgreement	1049
GetAgreementTerms	1055
SearchAgreements	1061
AWS Marketplace Metering API	1067
BatchMeterUsage	1069
MeterUsage	1075
RegisterUsage	1081
ResolveCustomer	1087
AWS Marketplace Entitlement API	1090
GetEntitlements	1091
AWS Marketplace Deployment API	1094
ListTagsForResource	1095
PutDeploymentParameter	1098
TagResource	1104
UntagResource	1107
AWS Marketplace Reporting API	1109
GetBuyerDashboard	1110
Data Types	1114
AWS Marketplace Catalog API	1118
AmiProductEntityIdFilter	1122
AmiProductFilters	1123
AmiProductLastModifiedDateFilter	1125
AmiProductLastModifiedDateFilterDateRange	1126
AmiProductSort	1128
AmiProductSummary	1130
AmiProductTitleFilter	1132
AmiProductVisibilityFilter	1134
BatchDescribeErrorDetail	1135
Change	1137
ChangeSetSummaryListItem	1140
ChangeSummary	1143
ContainerProductEntityIdFilter	1145

ContainerProductFilters	1146
ContainerProductLastModifiedDateFilter	1148
ContainerProductLastModifiedDateFilterDateRange	1149
ContainerProductSort	1151
ContainerProductSummary	1153
ContainerProductTitleFilter	1155
ContainerProductVisibilityFilter	1157
DataProductEntityIdFilter	1158
DataProductFilters	1159
DataProductLastModifiedDateFilter	1161
DataProductLastModifiedDateFilterDateRange	1162
DataProductSort	1164
DataProductSummary	1166
DataProductTitleFilter	1168
DataProductVisibilityFilter	1170
Entity	1171
EntityDetail	1173
EntityRequest	1175
EntitySummary	1177
EntityTypeFilters	1181
EntityTypeSort	1183
ErrorDetail	1185
Filter	1187
OfferAvailabilityEndDateFilter	1189
OfferAvailabilityEndDateFilterDateRange	1190
OfferBuyerAccountsFilter	1192
OfferEntityIdFilter	1193
OfferFilters	1194
OfferLastModifiedDateFilter	1197
OfferLastModifiedDateFilterDateRange	1198
OfferNameFilter	1200
OfferProductIdFilter	1202
OfferReleaseDateFilter	1203
OfferReleaseDateFilterDateRange	1204
OfferResaleAuthorizationIdFilter	1206
OfferSort	1208

OfferStateFilter	1210
OfferSummary	1211
OfferTargetingFilter	1214
ResaleAuthorizationAvailabilityEndDateFilter	1215
ResaleAuthorizationAvailabilityEndDateFilterDateRange	1217
ResaleAuthorizationCreatedDateFilter	1219
ResaleAuthorizationCreatedDateFilterDateRange	1221
ResaleAuthorizationEntityIdFilter	1223
ResaleAuthorizationFilters	1224
ResaleAuthorizationLastModifiedDateFilter	1227
ResaleAuthorizationLastModifiedDateFilterDateRange	1228
ResaleAuthorizationManufacturerAccountIdFilter	1230
ResaleAuthorizationManufacturerLegalNameFilter	1232
ResaleAuthorizationNameFilter	1234
ResaleAuthorizationOfferExtendedStatusFilter	1236
ResaleAuthorizationProductIdFilter	1237
ResaleAuthorizationProductNameFilter	1239
ResaleAuthorizationResellerAccountIDFilter	1241
ResaleAuthorizationResellerLegalNameFilter	1243
ResaleAuthorizationSort	1245
ResaleAuthorizationStatusFilter	1247
ResaleAuthorizationSummary	1248
SaaSProductEntityIdFilter	1252
SaaSProductFilters	1253
SaaSProductLastModifiedDateFilter	1255
SaaSProductLastModifiedDateFilterDateRange	1256
SaaSProductSort	1258
SaaSProductSummary	1260
SaaSProductTitleFilter	1262
SaaSProductVisibilityFilter	1264
Sort	1265
Tag	1267
AWS Marketplace Agreement API	1268
AcceptedTerm	1270
Acceptor	1273
AgreementViewSummary	1274

ByolPricingTerm	1277
ConfigurableUpfrontPricingTerm	1278
ConfigurableUpfrontPricingTermConfiguration	1280
ConfigurableUpfrontRateCardItem	1282
Constraints	1284
Dimension	1286
DocumentItem	1288
EstimatedCharges	1290
Filter	1292
FixedUpfrontPricingTerm	1294
FreeTrialPricingTerm	1296
GrantItem	1298
LegalTerm	1300
PaymentScheduleTerm	1302
ProposalSummary	1304
Proposer	1306
RateCardItem	1307
RecurringPaymentTerm	1309
RenewalTerm	1311
RenewalTermConfiguration	1313
Resource	1314
ScheduleItem	1316
Selector	1318
Sort	1320
SupportTerm	1322
UsageBasedPricingTerm	1324
UsageBasedRateCardItem	1326
ValidationExceptionField	1327
ValidityTerm	1329
AWS Marketplace Metering API	1330
Tag	1331
UsageAllocation	1333
UsageRecord	1335
UsageRecordResult	1337
AWS Marketplace Entitlement API	1338
Entitlement	1339

EntitlementValue	1341
AWS Marketplace Deployment API	1342
DeploymentParameterInput	1343
AWS Marketplace Reporting API	1344
Common Parameters	1345
Common Errors	1348
Document history	1350
Release notes for AWS Marketplace Discovery API	1354
Discovery API release notes for 2024	1354
Discovery API release notes for 2022	1355

Welcome to the AWS Marketplace API Reference

This API reference describes how AWS Marketplace sellers can use service APIs to integrate and manage product lifecycles and offers. It also describes how buyers can use service APIs to discover third-party software, data, and services as well as how to govern private marketplaces.

About the AWS Marketplace APIs

The following sections provide general information on the APIs.

Topics

- [Using the AWS Marketplace API](#)
- [Supported AWS Regions for the AWS Marketplace API](#)
- [Permissions for the AWS Marketplace API](#)
- [Quotas for the AWS Marketplace API](#)
- [Logging for the AWS Marketplace API](#)
- [Notifications for the AWS Marketplace API](#)

Using the AWS Marketplace API

The following sections provide information about using the AWS Marketplace APIs.

Topics

- [Using the AWS Marketplace Catalog API](#)
- [Using the AWS Marketplace Agreement API](#)
- [Using the AWS Marketplace Metering API](#)
- [Using the AWS Marketplace Entitlement API](#)
- [Using the AWS Marketplace Deployment API](#)
- [Using the AWS Marketplace Discovery API](#)

Using the AWS Marketplace Catalog API

The AWS Marketplace Catalog API service provides an API interface to manage AWS Marketplace for your AWS organization or AWS account. For approved sellers, you can manage your products programmatically, including the self-service publishing capabilities on the [AWS Marketplace Management Portal](#). For private marketplace administrators, you can manage your private marketplace programmatically.

With Catalog API actions, you can view and update your existing product programmatically. You can automate your product update process by integrating the AWS Marketplace Catalog API

with your AWS Marketplace product build or deployment pipelines. You can also create your own applications on top of the Catalog API to manage your products in AWS Marketplace. You can manage the products that users in your AWS account or AWS organization can see and purchase through your private marketplace.

The AWS Marketplace Catalog API service provides standard AWS API functionality. You can directly use the REST API actions described in [Actions](#), or you can use an AWS SDK to access an API that's tailored to the programming language or platform that you're using. For more information about AWS application development, see [Getting Started with AWS](#). For more information about using AWS SDKs, see [AWS SDKs](#).

Catalog API entities

AWS Marketplace entities are containers of data which serve different business purposes, such as a product or offer. Entities are categorized by types. Each entity type encapsulates data related to a specific business domain (for example, a product or a seller account).

To simplify this paradigm, entities are designed with some level of commonality in their structures. As a result, introducing a new business domain doesn't require that you learn a completely new structure.

General structure

The general structure of any entity is:

- A named type with a version
- An identifier for the specific instance of the type
- One or more facets that include the attributes of the entity

Type versioning

Every named type has a type and version associated with it, for example, *Entity*Product@1.0. The *type* (*Entity*Product) represents the classification of the content. The *version* (1.0) represents the structure of *Entity*Product.

The version gives you details about the structure of the entity. The following describes when a version will be changed:

- Existing entities won't be restructured without changing the version. Additions of optional new fields will result in a minor version update.

- Any feature that fundamentally changes the structure of a type leads to a major version update. Examples include:
 - Removing a field
 - Renaming a field (different name for the same semantic)
 - Changing the semantic of an existing field (for example, changing the expected type)
- A major version update can retain a subset of facets from the previous version.
- Users are provided notifications and documentation for new versions.

Identifier

Each entity represents a unique *thing* within a business domain. To identify the unique thing, we use an identifier associating an `EntityId` with a `RevisionId`, for example, `prod-ad8EXAMPLE651@3`. In this example, the `EntityId` is `prod-ad8EXAMPLE651` and the `RevisionId` is `3`. Every successful change request to the entity will update the revision.

The following are important details about the identifier:

- Each entity is uniquely identified by its `EntityId`, which is the key to globally distinguish one entity from another.
- Each published revision of an entity has a `RevisionId`. The `RevisionId`, along with the `EntityId`, distinguish one published revision from another.
- AWS Marketplace generates `EntityIds` and `RevisionIds`.

You can use the `DescribeEntity` action to find the details and the Identifier with the most recent `revisionId`.

The `RevisionId` is an optional part of requests to `StartChangeSet` (see [Working with change sets](#)). If you include a `RevisionId`, then the request to `StartChangeSet` will fail with a `ValidationException` if the `RevisionId` is not the latest revision of the entity. This allows you to implement optimistic locking in your application.

Note

When you include a `RevisionId` that is not the latest revision, the `ValidationException` message includes the latest `RevisionId`.

If you omit the `RevisionId`, the request is performed on the latest revision of the entity automatically.

Warning

Two requests to change the same object could result with one request overwriting the changes of the other request, as the second request rewrites data changed by the first request. Using `RevisionIds` in your requests prevents this issue by not allowing a change to an earlier revision to overwrite the current revision.

Facets

A facet is a logical grouping of attributes. An entity usually includes several facets which represent different aspects of the entity. The attributes within a facet have the following properties:

- Each attribute has a unique name within the scope of the container it belongs to.
- Attributes can be of a simple type (string, integer, or floating number).
- Attributes can be of a complex type (container/structure or array).

Entity type

The entity type defines what the entity represents. An entity can be a seller product in AWS Marketplace or a private marketplace. For more information, see [Working with seller products](#) and [Working with a private marketplace](#).

Working with change sets

When using the Catalog API, requests are created and updated through entities and completed by using change requests. Every change specifies the entity to be changed, the type of change to be performed, and details of the change. The type of change to be performed is called a `ChangeType`. A collection of `ChangeTypes` is called a `ChangeSet`.

There are four actions that allow you to work with change sets:

- `StartChangeSet` – Requests a set of changes. The changes are added to a queue and processed. For more information, see [Working with seller products](#) and [Working with a private marketplace](#).

- **DescribeChangeSet** – Gets the details of a set of changes, including the status of the request. The statuses include:
 - **PREPARING** – Getting ready to apply the changes.
 - **APPLYING** – In the process of making the requested changes.
 - **SUCCEEDED** – Request was completed successfully.
 - **CANCELLED** – Request was canceled by the user.
 - **FAILED** – Request was completed unsuccessfully. Further details are available in the response.
- **ListChangeSets** – Gets a list of the change sets that are currently in process.
- **CancelChangeSet** – Requests a change set be canceled. Changes can only be canceled while in the **PREPARING** status.

A typical workflow is to request a change with **StartChangeSet**, and then use the returned **ChangeSetId** to poll the **DescribeChangeSet** action until the change is complete.

The following is an example of the **DescribeChangeSet** response.

```
{
  "ChangeSet":
  [
    {
      "ChangeName": "myChangeName",
      "ChangeType": "UpdateInformation",
      "Details": "{ \"ProductTitle\": \"My Product Title\", \"ShortDescription\": \"My product short description.\", \"LongDescription\": \"My product longer description.\", \"Sku\": \"123example456\", \"SupportDescription\": \"Need help? Contact our experts at support@example.com\\n\\nYour purchase includes 24x7 support.\", \"Categories\": [ \"Operating Systems\", \"Network Infrastructure\", \"Application Development\" ]}",
      "DetailsDocument":
      {
        "ProductTitle": "My Product Title",
        "ShortDescription": "My product short description.",
        "LongDescription": "My product longer description.",
        "Sku": "123example456",
        "SupportDescription": "Need help? Contact our experts at support@example.com\\n\\nYour purchase includes 24x7 support.",
        "Categories":
        [
          "Operating Systems",
```



```
        "Network Infrastructure",
        "Application Development"
    ]
},
"Entity":
{
    "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
    "Type": "AmiProduct@1.0"
},
"ErrorDetailList":
[]
}
],
"ChangeSetArn": "arn:aws:aws-marketplace:[exampleARN]",
"ChangeSetId": "example123456789012abcdef",
"ChangeSetName": "myChangeSetName",
"EndTime": "2023-03-03T00:00:00Z",
"FailureCode": null,
"FailureDescription": null,
"StartTime": "2023-03-02T00:00:00Z",
"Status": "SUCCEEDED"
}
```

Note

When polling or working with change sets programmatically, you must adhere to the service limits. For more information, see [Service quotas for AWS Marketplace Catalog API](#).

After your change is complete, you can use `ListEntities` to find the entity that you created or modified (and its associated `EntityID`). You can then use `DescribeEntity` with the `EntityID` to get details about it.

For more information about working with change requests in the console for sellers, see [Creating a change request](#) in the *AWS Marketplace Seller Guide*.

Making multiple change requests simultaneously

Within a **single change set**, you can bundle all change types and they are run together. Catalog API is built to make multiple changes simultaneously to provide the best performance. Sellers and Channel Partners can invoke changes with multiple `ChangeTypes` bundled into a `ChangeSet`. You

can invoke multiple changes on single or different entities in the same `ChangeSet`. Catalog API evaluates which order the changes need to be applied and makes those changes.

However, if the requests are made as **separate change sets**, AWS Marketplace can't initiate conflicting change requests on the same product. In these cases, AWS Marketplace returns a `ResourceInUseException` error.

- For modifying AMI and container products, most changes can be made without error, with the following exceptions:
 - If two requests are the same `ChangeType` on the same product, the second request returns an error.
 - If one request is to update the version information, and the other request is to restrict or add a version, then the second request returns an error.
 - If a request is `PREPARING`, another request can be made on the same product. However, a change that is currently `APPLYING` may block other requests, returning an error.
- For other product types and private marketplaces, you can only have a single request for a product at a time. If a different request to update the same product is made while a first request is ongoing, the second returns an error.
- If there is a request for any product that is pending with the AWS Marketplace Seller Operations team, then any other requests on that product return an error.

If you receive a `ResourceInUseException` error for a change request, you can retry the request later. Depending on the state of the ongoing request, you can also cancel the first request, to allow the resubmitted second request to complete sooner.

Invoking multiple change types in one change set

You can use the Catalog API to combine and chain up to 20 changes in one `StartChangeSet` request targeting one or multiple different entities.

A typical use case is to create a `SaaSProduct@1.0` draft product, an `Offer@1.0` draft offer, and also filling in the metadata information of the product and offer. This is done by including the following four change types in one change set:

- `CreateProduct` on `SaaSProduct@1.0`

Specify the `ChangeName` parameter. Then, the product created in this change type can be referenced in the same change set by subsequent changes.

For example, `CreateProductChange`.

- `UpdateInformation` on the `SaaSProduct@1.0` created in the same change set

In the `Entity.Identifier` field, you can refer to the product created by `CreateProduct` change type using the change name in this format:

```
${ChangeName}.Entity.Identifier
```

For example, `$CreateProductChange.Entity.Identifier`.

- `CreateOffer` on `Offer@1.0` tied to the `SaaSProduct@1.0` created in the same change set

Specify the `ChangeName` parameter. Then, the product created in this change type can be referenced in the same change set by subsequent changes. For example, `CreateOfferChange`.

For the `ProductId` parameter in the payload of `CreateOffer` change type, you can also refer to the SaaS product created in `CreateProduct` change type by using `${ChangeName}.Entity.Identifier` syntax.

For example, `{"ProductId": "$CreateProductChange.Entity.Identifier"}`.

- `UpdateInformation` on the `Offer@1.0` created in the same change set

In the `Entity.Identifier` field, you can refer to the offer created by the `CreateOffer` change type using the change name in this format:

```
${ChangeName}.Entity.Identifier
```

For example, `$CreateOfferChange.Entity.Identifier`.

The following is an example of a combined change set.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
```

```

    "Type": "SaaSProduct@1.0"
  },
  "ChangeName": "CreateProductChange",
  "DetailsDocument": {}
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "ChangeName": "UpdateProductInformationChange",
  "DetailsDocument": {
    "ProductTitle": "My Product Title",
    "ShortDescription": "My product short description.",
    "LongDescription": "My product longer description.",
    "Sku": "123example456",
    "LogoUrl": "https://s3.amazonaws.com/presigned-or-public-url-to-logo-stored-in-
s3",
    "VideoUrls": [
      "https://example.com"
    ],
    "Highlights": [
      "123example45"
    ],
    "AdditionalResources": "123example456",
    "SupportDescription": "Need help? Contact our experts at support@example.com \n
\nYour purchase includes 24x7 support.",
    "SupportResources": "123example456",
    "Categories": [
      "Operating Systems",
      "Network Infrastructure",
      "Application Development"
    ],
    "SearchKeywords": "123example456"
  }
},
{
  "ChangeType": "CreateOffer",
  "Entity": {
    "Type": "Offer@1.0"
  },
  "ChangeName": "CreateOfferChange",
  "DetailsDocument": {

```

```
    "ProductId": "$CreateProductChange.Entity.Identifier"
  }
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Name": "Offer created together with SaaSProduct",
    "Description": "Test offer created together with SaaSProduct in the same
Catalog API change set"
  }
}
]
```

Working with the Details attribute (Legacy)

Note

This section describes the legacy Details attribute in your change request, which requires additional formatting for your change details. We recommend using the alternative DetailsDocument attribute. It doesn't require additional formatting and the change details don't need to be changed. For examples of the DetailsDocument attribute, see [Working with seller products](#) and [Working with a private marketplace](#).

The Details attribute of the StartChangeSet operation is a string value. Its contents are JSON objects. To put a JSON object into a string attribute, you must convert the object to a single-line string by escaping all JSON control characters, and removing line breaks.

For example, if you are using the StartChangeSet operation with UpdateProcurementPolicy to disable requests from users in your private marketplace, make a request like the following.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
```

```

"ChangeSet": [
  {
    "ChangeType": "UpdateProcurementPolicy",
    "Details": "<string>",
    "Entity": {
      "Type": "Experience@1.0",
      "Identifier" : "exp-1234example@5"
    }
  }
]
}

```

In this case, the JSON object that you use for the `Details` attribute looks like the following (before conversion to a string).

```

{
  "Configuration": {
    "PolicyResourceRequests": "Deny"
  }
}

```

But the `Details` attribute requires a string, not JSON. After converting this JSON object to a single line string, it looks like the following.

```

"{\"Configuration\" : {\"PolicyResourceRequests\" : \"Deny\"}}"

```

With this string, you can create the full change set request, as follows.

```

POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateProcurementPolicy",
      "Details": "{ \"Configuration\" : { \"PolicyResourceRequests\" : \"Deny\" } }",
      "Entity": {
        "Type": "Experience@1.0",
        "Identifier" : "exp-1234example@5"
      }
    }
  ]
}

```

```
]
}
```

Generally, examples in this API reference show the JSON object already converted to a string. In some cases, more complicated samples with new lines are included to enhance understanding.

Automate converting JSON to a string

Converting a JSON object to a string can be automated using tools such as [jq](#), a lightweight command-line JSON processor. The following example shows using `jq` to convert a JSON object to a string that can be used in the `Details` attribute.

```
DETAILS_JSON='{
  "ProductTitle": "My Product Title",
  "ShortDescription": "My product short description.",
  "LongDescription": "My product long description."
}';

DETAILS_JSON_STRING="$(echo "${DETAILS_JSON}" | jq 'tostring');"
```

If you echo `"${DETAILS_JSON_STRING}"`, the result is the following string with JSON properly escaped: `{"ProductTitle": "My Product", "ShortDescription": "My product short description.", "LongDescription": "My product long description."}`

Using DescribeEntity to get information about your entities

You can programmatically get information about your existing entities, including products and private marketplace, through the Catalog API.

The `ListEntities` action returns a list of entities. Then, you can use the `DescribeEntity` action to get details about an individual entity. This can be directly useful, for example, to catalog the products you sell. It can also be useful when updating entities, because you can get the current state of the entity before updating just the parts that you want to update.

The following example shows using `ListEntities` to get a list of container products, and then using `DescribeEntity` to get information about one of the specific products.

```
POST /ListEntities HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
```

```
"EntityType": "ContainerProduct"
}
```

For the entity type, you must use the entity type without the version. It returns all entities of that type (and doesn't filter on version).

Here is a sample of the response to the `ListEntities` action.

```
{
  "EntitySummaryList": [
    {
      "Name": "Container Product 1",
      "EntityType": "ContainerProduct",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12",
      "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
      "LastModifiedDate": "2021-03-01T00:00:00Z",
      "Visibility": "Public"
    },
    {
      "Name": "Container Product 2",
      "EntityType": "ContainerProduct",
      "EntityId": "example2-abcd-1234-5ef6-7890abcdef12",
      "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
      "LastModifiedDate": "2021-03-02T00:00:00Z",
      "Visibility": "Public"
    }
  ],
  "NextToken": "exampleabcdef12345..."
}
```

To get the details of one of these products, use the `DescribeEntity` action. The following example shows how to get details about the first product returned above.

```
GET /DescribeEntity?catalog=AWSMarketplace&entityId=example1-abcd-1234-5ef6-7890abcdef12 HTTP/1.1
```

The following shows the response to `DescribeEntity`.

```
{
  "EntityType": "ContainerProduct@1.0",
  "EntityIdentifier": "example1-abcd-1234-5ef6-7890abcdef12@9",
  "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
}
```



```

    "LastModifiedDate": "2021-03-02T20:19:14Z",
    "Details": "{\\"Versions\\": [{\\"Id\\": \\"example2-0000-aaaa-5ef6-7890abcdef12\\",
    \\"ReleaseNotes\\": \\"My release notes\\", \\"UpgradeInstructions\\": \\"N/A\\", \\"VersionTitle
    \\": \\"1.0\\", \\"CreationDate\\": \\"2021-03-02T00:00:00.000Z\\", \\"Sources\\": [{\\"Type
    \\": \\"DockerImages\\", \\"Id\\": \\"example3-1111-bbbb-5ef6-7890abcdef12\\", \\"Images\\":
    [\\"111122223333.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-repo-1:some-
    tag\\"], \\"Compatibility\\": {\\"Platform\\": \\"Linux\\"}], \\"DeliveryOptions\\": [{\\"Id\\":
    \\"example4-2222-cccc-2222-cccccccccccc\\", \\"Type\\": \\"ElasticContainerRegistry\\",
    \\"SourceId\\": \\"example3-1111-bbbb-5ef6-7890abcdef12\\", \\"Title\\": \\"New delivery
    option 1\\", \\"ShortDescription\\": \\"Delivery option 1\\", \\"isRecommended\\": false,
    \\"Compatibility\\": {\\"AWSservices\\": [\\"ECS\\", \\"EKS\\"}], \\"Instructions\\": {\\"Usage\\":
    \\"test\\"}, \\"Recommendations\\": {\\"AdditionalArtifacts\\": []}, \\"Visibility\\": \\"Limited
    \\"}]}], \\"Description\\": {\\"Highlights\\": [\\"Some highlight\\"], \\"LongDescription
    \\": \\"Description of my product\\", \\"ProductCode\\": \\"123456789012abcdef1234567\\",
    \\"Manufacturer\\": null, \\"Visibility\\": \\"Limited\\", \\"AssociatedProducts\\": null, \\"Sku
    \\": null, \\"SearchKeywords\\": [\\"some keyword\\"], \\"ProductTitle\\": \\"Container Product 1\\",
    \\"ShortDescription\\": \\"Description of my product\\", \\"Categories\\": [\\"Operating Systems
    \"]}, \\"PromotionalResources\\": {\\"LogoUrl\\": \\"https://awsmp-logos.s3.amazonaws.com/
    PLACEHOLDER_Logo_for_Containers_products.png\\", \\"AdditionalResources\\": [], \\"Videos
    \\": []}, \\"SupportInformation\\": {\\"Description\\": \\"Description of support information.
    \\", \\"Resources\\": []}, \\"RegionAvailability\\": {\\"Regions\\": [\\"ap-south-1\\", \\"eu-
    west-3\\", \\"eu-north-1\\", \\"eu-west-2\\", \\"eu-west-1\\", \\"ap-northeast-2\\", \\"ap-
    northeast-1\\", \\"me-south-1\\", \\"ca-central-1\\", \\"sa-east-1\\", \\"ap-east-1\\", \\"ap-
    southeast-1\\", \\"ap-southeast-2\\", \\"eu-central-1\\", \\"us-east-1\\", \\"us-east-2\\", \\"us-
    west-1\\", \\"us-west-2\\"], \\"FutureRegionSupport\\": null}, \\"Repositories\\": [{\\"Url\\":
    \\"111122223333.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-repo-1\\", \\"Type\\":
    \\"ECR\\"}]}]",
    "DetailsDocument":
    {
      "Versions":
      [
        {
          "Id": "example2-0000-aaaa-5ef6-7890abcdef12",
          "ReleaseNotes": "My release notes",
          "UpgradeInstructions": "N/A",
          "VersionTitle": "1.0",
          "CreationDate": "2021-03-02T00:00:00.000Z",
          "Sources":
          [
            {
              "Type": "DockerImages",
              "Id": "example3-1111-bbbb-5ef6-7890abcdef12",
              "Images":
              [

```

```
    "111122223333.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-
repo-1:some-tag"
  ],
  "Compatibility":
  {
    "Platform": "Linux"
  }
},
"DeliveryOptions":
[
  {
    "Id": "example4-2222-cccc-2222-cccccccccccc",
    "Type": "ElasticContainerRegistry",
    "SourceId": "example3-1111-bbbb-5ef6-7890abcdef12",
    "Title": "New delivery option 1",
    "ShortDescription": "Delivery option 1",
    "isRecommended": false,
    "Compatibility":
    {
      "AWServices":
      [
        "ECS",
        "EKS"
      ]
    },
    "Instructions":
    {
      "Usage": "test"
    },
    "Recommendations":
    {
      "AdditionalArtifacts":
      []
    },
    "Visibility": "Limited"
  }
]
},
"Description":
{
  "Highlights":
  [
```

```
    "Some highlight"
  ],
  "LongDescription": "Description of my product",
  "ProductCode": "123456789012abcdef1234567",
  "Manufacturer": null,
  "Visibility": "Limited",
  "AssociatedProducts": null,
  "Sku": null,
  "SearchKeywords":
  [
    "some keyword"
  ],
  "ProductTitle": "Container Product 1",
  "ShortDescription": "Description of my product",
  "Categories":
  [
    "Operating Systems"
  ]
},
"PromotionalResources":
{
  "LogoUrl": "https://awsmp-logos.s3.amazonaws.com/
PLACEHOLDER_Logo_for_Containers_products.png",
  "AdditionalResources":
  [],
  "Videos":
  []
},
"SupportInformation":
{
  "Description": "Description of support information.",
  "Resources":
  []
},
"RegionAvailability":
{
  "Regions":
  [
    "ap-south-1",
    "eu-west-3",
    "eu-north-1",
    "eu-west-2",
    "eu-west-1",
    "ap-northeast-2",
```

```
    "ap-northeast-1",
    "me-south-1",
    "ca-central-1",
    "sa-east-1",
    "ap-east-1",
    "ap-southeast-1",
    "ap-southeast-2",
    "eu-central-1",
    "us-east-1",
    "us-east-2",
    "us-west-1",
    "us-west-2"
  ],
  "FutureRegionSupport": null
},
"Repositories":
[
  {
    "Url": "111122223333.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-
repo-1",
    "Type": "ECR"
  }
]
}
}
```

Note

The `DetailsDocument` attribute contains the entity details as a JSON object. The legacy `Details` attribute contains the same JSON object as a string.

Using the AWS Marketplace Agreement API

AWS Marketplace is a curated digital catalog that customers can use to find, buy, deploy, and manage third-party software, data, and services to build solutions and run their businesses. The AWS Marketplace Agreement Service provides an API interface that helps AWS Marketplace sellers manage their agreements, including listing, searching, and filtering agreements.

Using the AWS Marketplace Metering API

This reference provides descriptions of the AWS Marketplace Metering Service API. AWS Marketplace sellers can use this API to submit data for custom usage dimensions. For more information about the necessary permissions to use this API, see [AWS Marketplace metering and entitlement API permissions](#) in the *AWS Marketplace Seller Guide*.

Submitting metering records

[MeterUsage](#)

- Submits the metering record for an AWS Marketplace product
- Called from: Amazon Elastic Compute Cloud (Amazon EC2) instance or a container running on either Amazon Elastic Kubernetes Service (Amazon EKS) or Amazon Elastic Container Service (Amazon ECS)
- Supported product types: Amazon Machine Images (AMIs) and containers
- Vendor-metered tagging: supported allocation tagging

[BatchMeterUsage](#)

- Submits the metering record for a set of customers. AWS CloudTrail captures BatchMeterUsage API calls. Use CloudTrail to verify that the software as a subscription (SaaS) metering records that you sent are accurate by searching for records using the eventName of BatchMeterUsage. You can also use CloudTrail to audit records over time. For more information, see [CloudTrail concepts](#).
- Called from: SaaS applications
- Supported product type: SaaS
- Vendor-metered tagging: supports allocation tagging

Accepting new customers

[ResolveCustomer](#)

- Resolves the registration token that the buyer submits through the browser during the registration process. Obtains a CustomerIdentifier along with the CustomerAWSAccountId and ProductCode.
- Called from: SaaS application during the registration process

- Supported product type: SaaS
- Vendor-metered tagging: not applicable

Entitlement and metering for paid container products

[RegisteredUsage](#)

- Provides software entitlement and metering. Paid container software products sold through AWS Marketplace must integrate with the AWS Marketplace Metering Service and call the `RegisterUsage` operation. Free and Bring Your Own License model (BYOL) products for Amazon ECS or Amazon EKS aren't required to call `RegisterUsage`. However, you can do so if you want to receive usage data in your seller reports. For more information about using the `RegisterUsage` operation, see [Container-based products on AWS Marketplace](#).
- Called from: paid container software products
- Supported product type: containers
- Vendor-metered tagging: not applicable

Using the AWS Marketplace Entitlement API

The AWS Marketplace Entitlement Service is used to determine the entitlement of a customer to a given product. An entitlement represents capacity in a product owned by the customer. For example, a customer might own some number of users or seats in an SaaS application or some amount of data capacity in a multi-tenant database.

Using the AWS Marketplace Deployment API

The AWS Marketplace Deployment Service supports Quick Launch, which is a deployment option for software as a service (SaaS) products. Quick Launch can help you configure, deploy, and launch products. The AWS Marketplace Deployment Service provides sellers with a secure method for passing deployment parameters (for example, API keys and external IDs) to buyers.

Note

For information about Quick Launch for sellers, see [Configure Quick Launch](#).

Using the AWS Marketplace Discovery API

The Discovery API can help direct your customers to the most relevant AWS Marketplace pages. Use the Discovery API for the following:

- Retrieve listing information, such as long and short product descriptions, marketplace categories, badges, media, pricing models, pricing units, publishers, and reviews.
- Populate your website, platform, and private marketplace with AWS Marketplace listings.
- Create custom views of AWS Marketplace listings for customers.

Note

The Discovery API is available to customers who have been approved by the AWS Marketplace team. To request access, contact the [AWS Marketplace Seller Operations team](#). If the team approves your access, you will receive a unique integration ID and an SDK to integrate with and call the Discovery API.

Supported AWS Regions for the AWS Marketplace API

The following sections provide information about supported AWS Regions.

Topics

- [Supported AWS Regions for the AWS Marketplace Catalog API](#)
- [Supported AWS Regions for the AWS Marketplace Agreement API](#)
- [Supported AWS Regions for the AWS Marketplace Metering API](#)
- [Supported AWS Regions for the AWS Marketplace Entitlement API](#)
- [Supported AWS Regions for the AWS Marketplace Deployment API](#)

Supported AWS Regions for the AWS Marketplace Catalog API

You can access the AWS Marketplace Catalog API from the US East (N. Virginia) AWS Region with the following endpoint.

```
catalog.marketplace.us-east-1.amazonaws.com
```

Supported AWS Regions for the AWS Marketplace Agreement API

You can access the AWS Marketplace Agreement Service from the US East (N. Virginia) AWS Region with the following endpoint.

```
agreement-marketplace.us-east-1.amazonaws.com
```

Supported AWS Regions for the AWS Marketplace Metering API

BatchMeterUsage Region support

BatchMeterUsage is supported in the following AWS Regions

Commercial Regions

eu-north-1, me-south-1, ap-south-1, eu-west-3, ap-southeast-3, us-east-2, af-south-1, eu-west-1, me-central-1, eu-central-1, sa-east-1, ap-east-1, ap-south-2, us-east-1, ap-northeast-2, ap-northeast-3, eu-west-2, ap-southeast-4, eu-south-1, ap-northeast-1, us-west-2, us-west-1, ap-southeast-1, ap-southeast-2, il-central-1, ca-central-1, eu-south-2, eu-central-2

China Regions

cn-northwest-1

MeterUsage Region support for Amazon EC2

MeterUsage is supported in the following AWS Regions

Commercial Regions

eu-north-1, me-south-1, ap-south-1, eu-west-3, ap-southeast-3, us-east-2, af-south-1, eu-west-1, me-central-1, eu-central-1, sa-east-1, ap-east-1, ap-south-2, us-east-1, ap-northeast-2, ap-northeast-3, eu-west-2, ap-southeast-4, eu-south-1, ap-northeast-1, us-west-2, us-west-1, ap-southeast-1, ap-southeast-2, il-central-1, ca-central-1, eu-south-2, eu-central-2

AWS GovCloud (US) Regions

us-gov-east-1, us-gov-west-1

Note

To add AWS Regions for metering, contact [AWS Marketplace Seller Operations](#).

MeterUsage Region support for Amazon ECS and Amazon EKS

Amazon ECS and Amazon EKS is supported in the following AWS Regions

Commercial Regions

us-east-1, us-east-2, us-west-1, us-west-2, eu-west-1, eu-central-1, eu-west-2, eu-west-3, eu-north-1, ap-east-1, ap-southeast-1, ap-northeast-1, ap-southeast-2, ap-northeast-2, ap-south-1, ca-central-1, sa-east-1.

Note

MeterUsage is unavailable in AWS GovCloud (US) for Amazon ECS and Amazon EKS.

RegisterUsage Region support

RegisterUsage is supported in the following AWS Regions

Commercial Regions

eu-north-1, ap-south-1, eu-west-3, us-east-2, eu-west-1, eu-central-1, sa-east-1, ap-east-1, us-east-1, ap-northeast-2, eu-west-2, ap-northeast-1, us-west-2, us-west-1, ap-southeast-1, ap-southeast-2, ca-central-1

ResolveCustomer Region support

ResolveCustomer is supported in the following AWS Regions

Commercial Regions

eu-north-1, me-south-1, ap-south-1, eu-west-3, ap-southeast-3, us-east-2, af-south-1, eu-west-1, me-central-1, eu-central-1, sa-east-1, ap-east-1, ap-south-2, us-east-1, ap-northeast-2, ap-northeast-3, eu-west-2, ap-southeast-4, eu-south-1, ap-northeast-1, us-west-2, us-west-1, ap-southeast-1, ap-southeast-2, il-central-1, ca-central-1, eu-south-2, eu-central-2

China Regions

cn-northwest-1

Supported AWS Regions for the AWS Marketplace Entitlement API

You can access the AWS Marketplace Entitlement Service from the US East (N. Virginia) AWS Region with the following endpoint.

```
entitlement.marketplace.us-east-1.amazonaws.com
```

Supported AWS Regions for the AWS Marketplace Deployment API

You can access the AWS Marketplace Deployment Service from the US East (N. Virginia) AWS Region with the following endpoint.

```
deployment-marketplace.us-east-1.amazonaws.com
```

Permissions for the AWS Marketplace API

The following sections provide information about permissions.

Topics

- [Access control for the AWS Marketplace Catalog API](#)
- [Access control for the AWS Marketplace Agreement API](#)
- [Access control for the AWS Marketplace Metering API](#)
- [Access control for the AWS Marketplace Entitlement API](#)
- [Access control for the AWS Marketplace Deployment API](#)
- [Access control for the AWS Marketplace Discovery API](#)
- [Access control for the AWS Marketplace Reporting API](#)

Access control for the AWS Marketplace Catalog API

You can use the AWS Marketplace Catalog API to manage [a seller product in AWS Marketplace](#) or an [experience in a private marketplace](#). However, first make sure your user or role can access the API functionality that you want to call.

Use AWS Identity and Access Management (IAM) to create users and roles and assign policies that grant limited permissions to end users. The policies define the actions that the user or role can take on your resources through the AWS Marketplace Catalog API.

For example, you can define roles such as engineering, marketing, and pricing. Then, you can add a user in your organization to the engineering role. In that role, they might be granted permissions to initiate a change request to publish a new version of your seller product. However, the engineering role doesn't allow the user to list all change sets.

Note

To sell products on AWS Marketplace, your AWS account must be set up as a seller account. For more details about becoming an AWS Marketplace seller, see [Getting started as a seller](#) in the *AWS Marketplace Seller Guide*.

You can use AWS managed policies, or you can create your own IAM policies to have more granular control than what's available in AWS managed policies. For details about these approaches, see the following topics.

Topics

- [Allowing actions with AWS managed policies](#)
- [Allowing actions on all resources](#)
- [Allowing actions on specific resources](#)
- [Allowing actions with specific ChangeType condition key](#)
- [Allowing actions with specific aws:ResourceTag condition key](#)
- [Creating a custom IAM role](#)
- [Managing tags on resources](#)
- [Managing tags when requesting changes to resources](#)
- [Granting permission to manage tags on resources](#)
- [Granting permission to manage tags on resources only when those resources have specific tags](#)
- [Granting permission to create entities and change sets only with tags](#)

Allowing actions with AWS managed policies

You can use policies that are managed by AWS to grant permissions to your user or role.

To work with products that you sell on AWS Marketplace, you can use the `AWSMarketplaceSellerFullAccess` IAM managed policy, which has full access to the AWS

Marketplace Catalog API in addition to its other permissions. You can grant read-only access for the Catalog API with the `AWSMarketplaceSellerProductsReadOnly` policy. For more information, see [Controlling access to AWS Marketplace Management Portal](#), [Policies and permissions for AWS Marketplace sellers](#), and [AWS managed policies for AWS Marketplace sellers](#) in the *AWS Marketplace Seller Guide*.

To manage a private marketplace, you can use the `AWSPrivateMarketplaceAdminFullAccess` IAM managed policy, which has full access to create and edit the private marketplace for your account or AWS organization. For more information, see [Controlling access to AWS Marketplace subscriptions](#), [Creating a private marketplace administrator](#), and [AWS managed policies for AWS Marketplace buyers](#) in the *AWS Marketplace Buyer Guide*.

Alternatively, you can create your own IAM policies to have more granular control than is available in AWS managed policies. Use the following topics to create your own IAM policies.

Allowing actions on all resources

Resources are objects that the actions can act upon. Not every resource type can be specified with every action. Some resource types work with only certain actions. For more information, see [Actions, resources, and condition keys for the AWS Marketplace Catalog](#) in the *Service Authorization Reference*.

There are two resource types in the Catalog API:

- **Entity** – An entity is a [seller product in AWS Marketplace](#) or an [experience in a private marketplace](#).
- **ChangeSet** – A change set is returned each time you use Catalog API to make changes to an entity. The change set describes the requested changes and its status. A change set can be canceled if the status is in the `PREPARING` state.

To allow a user or role the permission to make changes to all entities in an AWS account, you can add the following IAM policy. With this policy, the user or role can use the `StartChangeSet` action on all resources ("*").

```
{
  "Statement": [
    {
      "Effect": "Allow",
```

```
    "Action": [  
      "aws-marketplace:StartChangeSet"  
    ],  
    "Resource": "*"    
  }  
]  
}
```

For information about all actions available for the Catalog API, see [Actions, resources, and condition keys for AWS Marketplace Catalog](#) in the *Service Authorization Reference*.

Allowing actions on specific resources

Note

Resource-level permissions and condition context keys for the `StartChangeSet` action are supported only when used with Catalog API. They are not supported when used with the [AWS Marketplace Management Portal](#).

Instead of allowing changes to all resources, you can use resource-level permissions to allow changes to specific resources.

For example, you can allow changes to a specific seller product in the AWS account instead of to all seller products. You do this by specifying the Amazon Resource Name (ARN) of the seller product in the `Resource` of the IAM policy.

Note

To specify granular, resource-level permissions with actions that create new change sets, you need to also include a `ChangeSet` ARN to the list of resources. The `ChangeSet` ARN must include the wildcard (`/*`) to match any new change set ID that's created as shown.

```
{  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  

```

```

    "aws-marketplace:StartChangeSet"
  ],
  "Resource": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
AmiProduct/example1-abcd-1234-5ef6-7890abcdef12",
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/*"
  ]
}
]
}

```

For another example, you can allow changes to a specific experience in a private marketplace instead of to all experiences. You do this by specifying the ARN of the experience in the Resource of the IAM policy.

Note

To specify granular, resource-level permissions with actions that create new change sets, you need to also include a ChangeSet ARN to the list of resources. The ChangeSet ARN must include the wildcard (/*) to match any new change set ID that's created as shown.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet"
      ],
      "Resource": [
        "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Experience/exp-
example12345",
        "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/*"
      ]
    }
  ]
}

```

Allowing actions with specific ChangeType condition key

Note

Resource-level permissions and condition context keys for the `StartChangeSet` action are supported only when used with Catalog API. They are not supported when used with the [AWS Marketplace Management Portal](#).

The Catalog API action `StartChangeSet` has several different change types. You can allow access to only specific change types.

For example, you might only want to allow changes to the metadata of the seller product, such as the product title, and not allow adding new product versions. In this example, the change type `UpdateInformation` allows changing the metadata of a seller product, including the title. For more information about the different change types, see [Working with seller products](#) and [Working with a private marketplace](#) in the *AWS Marketplace Catalog API Reference*.

To limit the action to one or multiple change types, specify the `ChangeType` in the condition keys. In the following example IAM policy, the condition operator `StringEquals` specifies that the action is only allowed if the `ChangeType` matches `UpdateInformation`. For more information about condition keys, see [Condition operators](#) in the *AWS Identity and Access Management User Guide*.

Note

To specify granular, resource-level permissions with actions that create new change sets, you need to also include a `ChangeSet` ARN to the list of resources. The `ChangeSet` ARN must include the wildcard (`/*`) to match any new change set ID that's created as shown.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet"
      ],
```

```

    "Resource": [
      "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/
example1-abcd-1234-5ef6-7890abcdef12",
      "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/*"
    ],
    "Condition": {
      "StringEquals": {
        "catalog:ChangeType": "UpdateInformation"
      }
    }
  }
]
}

```

Allowing actions with specific `aws:ResourceTag` condition key

Note

Resource-level permissions and condition context keys for the `StartChangeSet` action are supported only when used with Catalog API. They are not supported when used with the [AWS Marketplace Management Portal](#).

You can allow actions on a group of entities without having to keep updating the policy and specifying a possibly growing list of entity ARNs. You can do this with resource tagging. Adding tags to resources allows you to control access to those resources based on their tags. For example, you might want to allow describing a group of seller products without specifying individual ARNs for each seller product.

For example, the following IAM policy allows the `DescribeEntity` action on any entity resource ("`*`") that has a tag key of `product-team` and tag value of `team-xyz`.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:DescribeEntity"
      ],
      "Resource": "*",
      "Condition": {

```



```
        "StringEquals": {
            "aws:ResourceTag/product-team": "team-xyz"
        }
    }
}
]
```

You can also allow describing and canceling change sets that were created with specific tags.

For example, the following IAM policy allows the `DescribeChangeSet` and `CancelChangeSet` actions on any change set resource ("`*`") that has a tag key of `product-team` and tag value of `team-xyz`.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:CancelChangeSet"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/product-team": "team-xyz"
        }
      }
    }
  ]
}
```

Also, you can allow starting change sets on entities only when those entities have specific tags. For example, you can allow changes to seller products with specific tags.

For example, the following IAM policy allows the `StartChangeSet` action on any entity resource ("`*`") that has a tag key of `product-team` and tag value of `team-xyz`. In addition, the `TagResource` action is required so that when the change set is created, it's tagged with the same tag key and value.

Note

If your policy to allow the `StartChangeSet` action includes a condition to match against specific tags, the same policy must also include the `TagResource` action. This is because the policy condition must match both the tag on the entity and the tag on the newly created change set resulting from the change request. Thus, it requires the user or role to also have the permission to tag the newly created change set. For an example of starting a change set and tagging the change set, see [the section called "Example: Adding tags to an entity and change set during creation"](#).

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:TagResource"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/product-team": "team-xyz"
        }
      }
    }
  ]
}
```

Creating a custom IAM role

Customers who want to use a Resale Authorization ChangeType or a CPPO ChangeType need to create a custom AWS Identity and Access Management (IAM) role. This will support the creation of the Resale Authorization product lifecycle.

To create a custom IAM role

1. Sign in to the IAM console (<https://console.aws.amazon.com/iam/>).
2. Under **Access management**, choose **Policies**.
3. Choose **Create policy**.

4. For **Step 1: Specify permissions**,

- a. In the **Policy editor**, select the **JSON** button, and then add the following policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowResaleAuthorizationShareActionsRAMCreate",
      "Effect": "Allow",
      "Action": [
        "ram:CreateResourceShare",
        "ram:AssociateResourceShare"
      ],
      "Resource": [
        "arn:aws:ram:*:*:*"
      ],
      "Condition": {
        "StringLikeIfExists": {
          "ram:ResourceArn": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/*"
        },
        "StringEqualsIfExists": {
          "ram:RequestedResourceType": "aws-marketplace:Entity"
        }
      }
    },
    {
      "Sid": "AllowResaleAuthorizationShareActionsRAMAccept",
      "Effect": "Allow",
      "Action": [
        "ram:AcceptResourceShareInvitation",
        "ram:GetResourceShareInvitations",
        "ram:GetResourcePolicies",
        "ram:GetResourceShareAssociations"
      ],
      "Resource": [
        "arn:aws:ram:*:*:*"
      ]
    },
    {
      "Sid": "AllowResaleAuthorizationShareActionsMarketplace",
      "Effect": "Allow",
```

```

    "Action": [
      "aws-marketplace:PutResourcePolicy",
      "aws-marketplace:GetResourcePolicy",
      "aws-marketplace:DescribeEntity"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/
*"
  }
]
}

```

- b. Choose **Next**.
5. For **Step 2: Review and create**,
 - a. For **Policy details**, enter **FullResaleAuthorizationAccess** under **Policy name** and enter an optional **Description**.
 - b. Review the **Permissions defined in this policy**.
 - c. For **Add tags**, add tags (optional).
 - d. Choose **Create policy**.

You have created the FullResaleAuthorizationAccess policy.

6. Under **Access management**, choose **Roles**.
7. Choose **Create role**.
8. For **Step 1: Select trusted entity**,
 - a. For **Trusted entity type**, choose **Custom trust policy**.
 - b. Copy and paste the following custom trust policy into the JSON editor.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "resale-authorization.marketplace.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}

```

```
}
```

- c. Choose **Next**.
9. For **Step 2: Add permissions**,
 - a. Enter **FullResaleAuthorizationAccess** in the search bar.
 - b. Select the **FullResaleAuthorizationAccess** permission policy and then choose **Next**.
 10. For **Step 3: Name, review, and create**,
 1. For **Role details**, enter **FullResaleAuthorizationAccess** as the **Role name** and enter an optional **Description**.
 2. Under **Step 1: Select trusted entities**, ensure that the policy name you choose is attached to the role.
 3. Under **Step 2: Add permissions**, review the **Policy name**.
 4. Under **Step 3: Add tags**, add tags (optional).
 5. Choose **Create role**.

You have created the FullResaleAuthorizationAccess role.

Managing tags on resources

You can add, list, and remove tags from existing entities or change sets.

Add tags to resources

To add tags to an entity or change set, use the TagResource API action.

Request

```
POST /TagResource HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ResourceArn": "string",
  "Tags": [
    {
      "Key": "string",
```

```

    "Value": "string"
  }
  ...
]
}

```

Request parameters include:

- **Catalog (String) – (Required)** Must be `AWSMarketplace`.
- **ResourceArn (String) – (Required)** ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a [seller product in AWS Marketplace](#) or an [experience in a private marketplace](#).
- **Tags (Array of objects) – (Required)** A list of objects specifying each tag key and value. Number of objects allowed: 1–50.
 - **Key (String) – (Required)** Name of the tag.
 - **Regex pattern** – `^([\p{L}\p{Z}\p{N}_./+=\-\@]*)$`
 - **Character length** – 1–128
 - **Value (String) – (Required)** Value of the tag.
 - **Regex pattern** – `^([\p{L}\p{Z}\p{N}_./+=\-\@]*)$`
 - **Character length** – 0–256

Response

```
{ }
```

Remove tags from resources

To remove a tag or list of tags from an entity or change set, use the `UntagResource` API action.

Request

```

POST /UntagResource HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ResourceArn": "string",
  "TagKeys": [

```

```
    "string"  
    ...  
  ]  
}
```

Request parameters include:

- **Catalog (String) – (Required)** Must be `AWSMarketplace`.
- **ResourceArn (String) – (Required)** ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a [seller product in AWS Marketplace](#) or an [experience in a private marketplace](#).
- **Tags (Array of objects) – (Required)** A list of key names of tags to be removed. Number of strings allowed: 0–256.

Response

```
{}
```

List all tags on a resource

To list all tags that have been added to and not yet removed from a change set or entity, use the `ListTagsForResource` API action.

Request

```
POST /ListTagsForResource HTTP/1.1  
Content-type: application/json
```

```
{  
  "Catalog": "AWSMarketplace",  
  "ResourceArn": "string"  
}
```

Request parameters include:

- **Catalog (String) – (Required)** Must be `AWSMarketplace`.
- **ResourceArn (String) - (Required)** ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a [seller product in AWS Marketplace](#) or an [experience in a private marketplace](#).

Response

```
{
  "ResourceArn": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
    ...
  ]
}
```

Managing tags when requesting changes to resources

You can add tags when entities or change sets are created.

Example: Adding tags to a change set when creating a change set

The following is an example of a `StartChangeSet` request that updates the product metadata for a seller product. This request adds a tag to the change set that's created with this request by including the tag in the `ChangeSetTags` property.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "Details": "{\"ProductTitle\": \"My updated title\"}"
    }
  ],
  "ChangeSetTags": [
    {
      "Key": "product-team",
      "Value": "team-xyz"
    }
  ]
}
```



```
]
}
```

For more information about managing seller products, see [Working with seller products](#) in the *AWS Marketplace Catalog API Reference*.

Example: Adding tags to an entity and change set during creation

The following is an example of a `StartChangeSet` request that creates a private marketplace experience entity. The request adds tags to both the entity resource and change set resource created with this request by including the tags in the `EntityTags` and `ChangeSetTags` properties. With these tags, the permission policy of a user or role can be specified to only allow describing or canceling the change set this request creates or only allow creating further change sets on the entity this request creates. For more information, see [Granting permission to create entities and change sets only with tags](#).

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateExperience",
      "Entity": {
        "Type": "Experience@1.0"
      },
      "Details": "{\"Name\": \"ExamplePrivateMarketplace\"}",
      "EntityTags": [
        {
          "Key": "product-team",
          "Value": "team-xyz"
        }
      ]
    }
  ],
  "ChangeSetTags": [
    {
      "Key": "product-team",
      "Value": "team-xyz"
    }
  ]
}
```

```
}
```

For more information about managing a private marketplace, see [Working with a private marketplace](#) in the *AWS Marketplace Catalog API Reference*.

Granting permission to manage tags on resources

To allow a user or role to add, remove, and list tags on all entities or change sets, they need the following IAM policy.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:TagResource",
        "aws-marketplace:UntagResource",
        "aws-marketplace:ListTagsForResource"
      ],
      "Resource": "*"
    }
  ]
}
```

Granting permission to manage tags on resources only when those resources have specific tags

You can allow a user or role to add, remove, and list tags on entities or change sets that have specific tags. The following IAM policy allows those actions on any entity resource ("*") that has a tag key of product-team and tag value of team-xyz.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:TagResource",
        "aws-marketplace:UntagResource",
        "aws-marketplace:ListTagsForResource"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:tag:product-team": "team-xyz"
        }
      }
    }
  ]
}
```

```
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/product-team": "team-xyz"
      }
    }
  ]
}
```

Granting permission to create entities and change sets only with tags

Note

Resource-level permissions and condition context keys for the `StartChangeSet` action are supported only when used with Catalog API. They are not supported when used with the [AWS Marketplace Management Portal](#).

You can enforce tagging when entities or change sets are created. Add the following policy to allow the `StartChangeSet` and the `TagResource` actions, with a condition specifying the tag key matches `product-team` and the tag value matches `team-xyz`. This policy condition must match both the tag on the newly created entity and the tag on the newly created change set resulting from the creation request. For an example of tagging an entity on creation, see [the section called “Example: Adding tags to an entity and change set during creation”](#).

For existing entities, this policy also enforces tagging change sets when requesting changes to those entities. This also requires that the existing entity has this existing tag. This is because the policy condition must match both the tag on the existing entity and the newly created change set resulting from the change request. For an example of adding tags to change requests, see [the section called “Example: Adding tags to a change set when creating a change set”](#).

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:TagResource"
      ],
      "Resource": "*"
    }
  ]
}
```

```
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/product-team": "team-xyz"
      }
    }
  ]
}
```

Access control for the AWS Marketplace Agreement API

To manage agreements in AWS Marketplace using the Agreement Service, you must ensure that your AWS Identity and Access Management (IAM) policies and roles are set up. Users must have the following policies/permissions to allow them to carry out the actions:

- `DescribeAgreement` – Grants permission to users to obtain detailed metadata about any of their agreements.
- `GetAgreementTerms` – Grants permission to users to obtain details about the terms of an agreement.
- `SearchAgreements` – Grants permission to users to search through all their agreements.

Note

For more information about these permissions, see [Policies and permissions for AWS Marketplace sellers](#) in the *AWS Marketplace Seller Guide*.

Access control for the AWS Marketplace Metering API

To integrate the AWS Marketplace Metering API with AWS Marketplace, the service account must have a constrained AWS Identity and Access Management (IAM) policy. For more information, see [AWS Marketplace metering and entitlement API permissions](#).

Access control for the AWS Marketplace Entitlement API

To integrate the AWS Marketplace Entitlement API with AWS Marketplace, the service account must have a constrained AWS Identity and Access Management (IAM). For more information, see [AWS Marketplace metering and entitlement API permissions](#).

Access control for the AWS Marketplace Deployment API

To manage deployments in AWS Marketplace, you must ensure that you have the necessary AWS Identity and Access Management (IAM) roles and permissions.

Before calling the `PutDeploymentParameter` action, buyers must create the **AWSServiceRoleForMarketplaceDeployment** service-linked role. This provides AWS Marketplace with the permissions required to create, manage, and tag the necessary deployment parameter related resources in the buyer's account. Buyers create this role using prompts as they progress through the configuration process for any Quick Launch experience. For more information, see [Using roles to configure and launch products](#) in *AWS Marketplace Buyer Guide*.

To call `PutDeploymentParameter`, sellers must have IAM permissions for the following actions:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:PutDeploymentParameter",
        "aws-marketplace:TagResource"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

The `aws-marketplace:PutDeploymentParameter` action permits the user to call the `PutDeploymentParameter` API. The API also accepts an optional `tags` attribute. If the `tags` attribute is included in the request, the caller must also have permissions for `aws-marketplace:TagResource` on the relevant resource. For more information about creating users, see [Creating a user in your AWS account](#) in the *IAM User Guide*. For more information about creating and assigning policies, see [Changing permissions for an IAM user](#).

Access control for the AWS Marketplace Discovery API

To request access or ask questions about the Discovery API, contact the [AWS Marketplace Seller Operations team](#). If the team approves your access, you will receive a unique integration ID and an SDK to integrate with and call the Discovery API.

Access control for the AWS Marketplace Reporting API

You use the AWS Marketplace Reporting API to get the **Procurement insights** dashboard. The API supports the [GetBuyerDashboard](#) action.

To use the API, you must first create the `AWSServiceRoleForProcurementInsightsPolicy` service-linked role. The role does the following:

- Enables AWS Marketplace to access and describe the data for all the accounts in a buyer's organization.
- Gets the **Procurement insights** dashboard
- Enables you to register and deregister delegated administrators

Buyers create the role when they use the AWS Marketplace console to enable trusted access to the **Procurement insights** dashboard. For more information about that process, see [Activating the dashboard](#), in the *AWS Marketplace Buyer Guide*.

Important

- When using the API or the CLI, you must create the service-linked role before you enable trusted access to the dashboard.
- In addition to the service-linked role, you must enable all features for your organization, and you must belong to an administrator account. For more information, see the following topics in the *AWS Organizations User Guide*:
 - [Enabling all features for an organization with AWS Organizations](#)
 - [Tutorial: Creating and configuring an organization](#)
 - [Managing the management account with AWS Organizations](#)

The `AWSServiceRoleForProcurementInsightsPolicy` must have the following IAM permissions in order to call the [GetBuyerDashboard](#) action, and to register and deregister delegated administrators:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "CreateServiceLinkedRoleForProcurementInsights",
```

```

    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "arn:aws:iam::*:role/aws-service-role/procurement-
insights.marketplace.amazonaws.com/AWSServiceRoleForProcurementInsights*",
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": "procurement-
insights.marketplace.amazonaws.com"
      }
    }
  },
  {
    "Sid": "EnableAWSServiceAccessForProcurementInsights",
    "Effect": "Allow",
    "Action": [
      "organizations:EnableAWSServiceAccess"
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "organizations:ServicePrincipal": "procurement-
insights.marketplace.amazonaws.com"
      }
    }
  },
  {
    "Sid": "ManageDelegatedAdministrators",
    "Effect": "Allow",
    "Action": [
      "organizations:ListDelegatedAdministrators",
      "organizations:DeregisterDelegatedAdministrator",
      "organizations:RegisterDelegatedAdministrator"
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "organizations:ServicePrincipal": "procurement-
insights.marketplace.amazonaws.com"
      }
    }
  },
  {
    "Sid": "GetBuyerDashboardStatement",
    "Effect": "Allow",

```

```

    "Action": "aws-marketplace:GetBuyerDashboard",
    "Resource": "*"
  },
  {
    "Sid": "ViewOrganizationDetails",
    "Effect": "Allow",
    "Action": [
      "organizations:DescribeOrganization",
      "organizations:ListAWSServiceAccessForOrganization"
    ],
    "Resource": "*"
  }
]
}

```

For more information about creating policies, see [Policies and permissions in AWS Identity and Access Management](#), in the *IAM User Guide*.

For more information about the AWS Organizations actions that the policy uses, see the [AWS Organizations API reference](#).

Quotas for the AWS Marketplace API

The following sections provide information about service quotas.

Topics

- [Service quotas for AWS Marketplace Catalog API](#)
- [Service quotas for AWS Marketplace Agreement API](#)
- [Service quotas for AWS Marketplace Deployment API](#)

Service quotas for AWS Marketplace Catalog API

The AWS Marketplace Catalog API has the following quotas.

Request quotas

API operations	Request rate (per AWS account)
ListEntities	10 per second

API operations	Request rate (per AWS account)
DescribeEntity	20 per second
StartChangeSet	5 per second
ListChangeSets	5 per second
DescribeChangeSet	10 per second
CancelChangeSet	5 per second
TagResource	5 per second
UntagResource	5 per second
ListTagsForResource	5 per second
PutResourcePolicy	5 per second
GetResourcePolicy	5 per second
DeleteResourcePolicy	5 per second

Account quotas

Quota	Description
Maximum number of open StartChangeSet requests per account	250
Maximum number of Offers created or updated concurrently per account	20

Request history retention quotas

Description	Quota
Retention period for change requests. This applies after the end time of each change request.	90 days

Service quotas for AWS Marketplace Agreement API

Your AWS account has the following quotas related to the AWS Marketplace Agreement Service.

Request quotas

API operation	Request rate (per AWS account)
DescribeAgreement	5 per second
GetAgreementTerms	5 per second
SearchAgreements	5 per second

Service quotas for AWS Marketplace Deployment API

Your AWS account has the following quotas related to the AWS Marketplace Deployment Service.

Request quotas

Quota	Default	Description
Deployment parameter quota	10	The maximum number of deployment parameters per buyer and product combination
Deployment parameter update frequency	100	The maximum number of times you can update a deployment parameter per 24 hours

Quota	Default	Description
PutDeploymentParameter throttle quota	1	The maximum number of PutDeploymentParameter requests that you can make per second
ListTagsForResource throttle quota	20	The maximum number of ListTagsForResource requests that you can make per second
TagResource throttle quota	20	The maximum number of TagResource requests that you can make per second
UntagResource throttle quota	20	The maximum number of UntagResource requests that you can make per second
Deployment parameter SecretString name length	15,000	The maximum number of characters for the deployment parameter SecretString string
Deployment parameter name length	400	The maximum number of characters in a deployment parameter name
TagList quota	50	The maximum number of tags per deployment parameter request
ClientToken quota	64	The maximum number of characters for the ClientToken string

Logging for the AWS Marketplace API

The following sections provide information about logging.

Topics

- [Logging AWS Marketplace Catalog API calls with CloudTrail](#)
- [Logging AWS Marketplace Agreement API calls with CloudTrail](#)
- [Logging AWS Marketplace Deployment API calls with CloudTrail](#)
- [Logging AWS Marketplace Discovery API calls with CloudTrail](#)

Logging AWS Marketplace Catalog API calls with CloudTrail

The AWS Marketplace Catalog API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service. CloudTrail captures all calls to the Catalog API as events, including calls from the AWS Marketplace Management Portal.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request, the IP address from which the request was made, who made the request, when it was made, and additional details.

AWS Marketplace Catalog API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the AWS Marketplace Catalog API, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see [Viewing Events with CloudTrail Event History](#) in the *AWS CloudTrail User Guide*.

For an ongoing record of events in your AWS account, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see:

- [Overview for Creating a Trail](#)
- [CloudTrail Supported Services and Integrations](#)
- [Configuring Amazon SNS Notifications for CloudTrail](#)
- [Receiving CloudTrail Log Files from Multiple Regions](#)
- [Receiving CloudTrail Log Files from Multiple Accounts](#)

All AWS Marketplace Catalog API actions are logged by CloudTrail and are documented in this API Reference. For example, calls to the `StartChangeSet`, `DescribeChangeSet`, and `ListChangeSets` API actions generate entries in the CloudTrail log files. Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see [CloudTrail userIdentity Element](#) in the *AWS CloudTrail User Guide*.

Understanding AWS Marketplace catalog log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any specific order.

Note

These examples have been formatted for improved readability. In a CloudTrail log file, all entries and events are concatenated into a single line. In addition, this example has been limited to a single AWS Marketplace Catalog API entry. In a real CloudTrail log file, you see entries and events from multiple AWS services.

The following example shows a AWS Marketplace Catalog API log entry that demonstrates the `ListEntities` action:

```
[
  {
    "eventVersion": "1.05",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "ABCDEFGHIJKLMN0P12345",
      "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
      "accountId": "123456789010",
      "accessKeyId": "ABCDEFGHIJKLMN0P1234",
      "userName": "CloudTrailTestUser"
    },
    "eventTime": "2019-10-17T21:49:23Z",
    "eventSource": "marketplacecatalog.amazonaws.com",
    "eventName": "ListEntities",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
      "catalog": "AWSMarketplace",
      "entityType": "EntityProduct",
      "sort": {
        "sortBy": "LastUpdateTimeInMillis",
        "sortOrder": "DESC"
      },
      "maxResults": 20
    },
    "responseElements": null,
    "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789010"
  }
]
```

Logging AWS Marketplace Agreement API calls with CloudTrail

The Agreements API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Marketplace. CloudTrail captures API calls for the Agreements API as events. The calls captured include calls from the AWS Marketplace website, console, and other interfaces leveraging the Agreements API, as well as direct code calls to Agreements API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for the Agreements API. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to the Agreements API, the IP address from which the request was made, who made the request, when it was made, and additional details.

For more information about CloudTrail, see the [AWS CloudTrail User Guide](#).

Agreements API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the Agreements API, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see [Viewing events with CloudTrail Event history](#) in the *AWS CloudTrail User Guide*.

For an ongoing record of events in your AWS account, including events for the Agreements API, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- [Overview for creating a trail](#)
- [CloudTrail supported services and integrations](#)
- [Configuring Amazon SNS notifications for CloudTrail](#)
- [Receiving CloudTrail log files from multiple regions](#) and [Receiving CloudTrail log files from multiple accounts](#)

The `AcceptAgreementRequest`, `CancelAgreement`, `DescribeAgreement`, `GetAgreementTerms`, and `SearchAgreements` Agreements API actions are logged by CloudTrail.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.

- Whether the request was made by another AWS service.

For more information, see [CloudTrail userIdentity element](#) in the *AWS CloudTrail User Guide*.

Understanding Agreements API actions

The Agreements API is used to purchase software as a service (SaaS), server (including container), and professional services products on AWS Marketplace. It's also used to manage resulting agreements or subscriptions on AWS Marketplace.

Note

Purchases of machine learning or AWS Data Exchange products won't be logged by CloudTrail.

The `AcceptAgreementRequest` action is used when an AWS Identity and Access Management (IAM) user or role of an AWS account purchases an applicable product on AWS Marketplace. Similarly, the `CancelAgreement` action is used when an IAM user or role cancels their agreement or subscription. By monitoring CloudTrail logs in the Agreements API, buyers can monitor the most important purchase-related actions happening in their AWS account on AWS Marketplace.

The `DescribeAgreement` action is used when the customer specifically views meta data for a specific agreement. The `GetAgreementTerms` action is used when the terms of a particular agreement are viewed. The `SearchAgreements` action is used when an IAM user or role lists or filters out a subset of their agreements from the full list of all their agreements.

Note

The `AcceptAgreementRequest` and `CancelAgreement` actions are available to buyers but not sellers. However, the `DescribeAgreement`, `GetAgreementTerms`, and `SearchAgreements` actions can be used by both buyers and sellers.

Buyers can also identify the Agreement ID of the agreement from the CloudTrail log. For more information about the agreement, choose the **Manage subscriptions** tab in the AWS Marketplace console, where the Agreement ID is provided in the **Details** view. The Agreement ID can be found in

responseElements for the AcceptAgreementRequest API action and in requestParameters for the CancelAgreement API action.

Understanding Agreements API log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't display in a specific order.

The following example shows a CloudTrail log entry that demonstrates the AcceptAgreementRequest action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP123"
  },
  "eventTime": "2023-08-11T17:13:50Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "AcceptAgreementRequest",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Coral/Netty4",
  "requestParameters": {
    "agreementRequestId": "ar-6xbrddjzym594imkrrezrn5wa"
  },
  "responseElements": {
    "agreementId": "agmt-1lnrq6riwpg2tczhv378zknlc"
  },
  "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
  "eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management"
}
```

```
}
```

The following example shows a CloudTrail log entry that demonstrates the `CancelAgreement` action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP1234"
  },
  "eventTime": "2023-08-14T03:11:42Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "CancelAgreement",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Coral/Netty4",
  "requestParameters": {
    "agreementId": "agmt-enitbfqjebjmwmomzrucf032t"
  },
  "responseElements": null,
  "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
  "eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management"
}
```

The following example shows a CloudTrail log entry that demonstrates the `DescribeAgreement` action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudtrailTestUser",
```

```

    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP123",
  },
  "eventTime": "2023-10-30T22:45:24Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "DescribeAgreement",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36",
  "requestParameters": {
    "agreementId": "agmt-6wy4rhp7l6iyuu2jrcgd1shdi"
  },
  "responseElements": null,
  "requestID": "bEXAMPLE-347f-4c07-9645-cd2EXAMPLE61",
  "eventID": "dEXAMPLE-d891-42a5-8da6-1cdEXAMPLE34",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management",
}

```

The following example shows a CloudTrail log entry that demonstrates the `GetAgreementTerms` action.

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudtrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP123",
  },
  "eventTime": "2023-10-30T22:48:37Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "GetAgreementTerms",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36",
  "requestParameters": {

```

```
    "agreementId": "agmt-6wy4rhp716iyuu2jrcgd1shdi"
  },
  "responseElements": null,
  "requestID": "eEXAMPLE-fc57-4127-bbda-bc1EXAMPLE03",
  "eventID": "bEXAMPLE-5345-4634-8b58-925EXAMPLE3e",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management",
}
```

The following example shows a CloudTrail log entry that demonstrates the SearchAgreements action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHijklmnop12345",
    "arn": "arn:aws:iam::123456789010:user/CloudtrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHijklmnop123",
  },
  "eventTime": "2023-10-30T18:41:10Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "SearchAgreements",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36",
  "requestParameters": {
    "catalog": "AWSMarketplace",
    "filters": [
      {
        "name": "PartyType",
        "values": [
          "Proposer"
        ]
      },
      {
        "name": "ResourceType",
        "values": [
```

```
        "SaaSProduct"
      ]
    },
    {
      "name": "Status",
      "values": [
        "ACTIVE"
      ]
    },
    {
      "name": "AgreementType",
      "values": [
        "PurchaseAgreement"
      ]
    }
  ],
  "maxResults": 5
},
"responseElements": null,
"requestID": "fEXAMPLE-0aa6-4e42-8715-6a1EXAMPLE95",
"eventID": "0EXAMPLE-8ce8-4814-bcf1-636EXAMPLEb5",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789010",
"eventCategory": "Management",
}
```

Logging AWS Marketplace Deployment API calls with CloudTrail

The AWS Marketplace Deployment Service is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request, the IP address from which the request was made, who made the request, when it was made, and additional details.

AWS Marketplace Deployment Service information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the AWS Marketplace Deployment Service, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see [Viewing Events with CloudTrail Event History](#) in the *AWS CloudTrail User Guide*.

For an ongoing record of events in your AWS account, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see:

- [Overview for Creating a Trail](#)
- [CloudTrail Supported Services and Integrations](#)
- [Configuring Amazon SNS Notifications for CloudTrail](#)
- [Receiving CloudTrail Log Files from Multiple Regions](#)
- [Receiving CloudTrail Log Files from Multiple Accounts](#)

All Deployment Service actions are logged by CloudTrail and are documented in this API reference. For example, calls to the `PutDeploymentParameterAPI` action generates entries in the CloudTrail log files. Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see [CloudTrail userIdentity Element](#) in the *AWS CloudTrail User Guide*.

Understanding AWS Marketplace log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single

request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any specific order.

Note

These examples have been formatted for improved readability. In a CloudTrail log file, all entries and events are concatenated into a single line. In addition, this example has been limited to a single Deployment Service entry. In a real CloudTrail log file, you see entries and events from multiple AWS services.

The following example shows a Deployment Service log entry that demonstrates the `PutDeploymentParameter` action:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHIJKLMN0P12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHIJKLMN0P123"
  },
  "eventTime": "2023-11-16T16:32:48Z",
  "eventSource": "deployment-marketplace.amazonaws.com",
  "eventName": "PutDeploymentParameter",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "aws-sdk-java/2.20.162 Mac_OS_X/13.5.2 OpenJDK_64-
Bit_Server_VM/18.0.1+10-FR Java/18.0.1 vendor/Amazon.com_Inc. io/sync http/
URLConnection cfg/retry-mode/legacy",
  "requestParameters": {
    "productId": "prod-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "catalog": "AWSMarketplace",
    "clientToken": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "agreementId": "agmt-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "deploymentParameter": {
      "name": "PutDeploymentParameterCloudTrailTest-secret",
      "secretString": "****"
    }
  },
  "expirationDate": "2023-11-30T03:02:26.779241Z"
}
```

```
    }
  },
  "responseElements": {
    "agreementId": "agmt-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "deploymentParametersId": "dp-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "resourceArn": "arn:aws:aws-marketplace:us-east-1:123456789010:DeploymentParameter:catalogs/AWSMarketplace/products/prod-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1/dp-fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1"
  },
  "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
  "eventID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management"
}
```

Logging AWS Marketplace Discovery API calls with CloudTrail

The Discovery API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Marketplace. CloudTrail captures all API calls for the Discovery API as events. The calls captured include calls from the AWS Marketplace console and code calls to the Discovery API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for the Discovery API. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to the Discovery API, the IP address from which the request was made, who made the request, when it was made, and additional details.

For more information about CloudTrail, see the [AWS CloudTrail User Guide](#).

Discovery API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the Discovery API, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see [Viewing events with CloudTrail Event history](#) in the *AWS CloudTrail User Guide*.

For an ongoing record of events in your AWS account, including events for the Discovery API, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- [Overview for creating a trail](#)
- [CloudTrail supported services and integrations](#)
- [Configuring Amazon SNS notifications for CloudTrail](#)
- [Receiving CloudTrail log files from multiple regions](#) and [Receiving CloudTrail log files from multiple accounts](#)

All Discovery API actions are logged by CloudTrail and are documented in this API Reference. For example, calls to the `SearchListings`, `GetSearchFacets`, and `GetListingView` actions generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see [CloudTrail userIdentity element](#) in the *AWS CloudTrail User Guide*.

Understanding Discovery API log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the `SearchListings` action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP1234",
    "userName": "CloudTrailTestUser",
    "sessionContext": {
      "sessionIssuer": {},
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2022-09-01T20:33:21Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2022-09-01T20:33:58Z",
  "eventSource": "discovery-marketplace.amazonaws.com",
  "eventName": "SearchListings",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "12.34.567.890",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:91.0) Gecko/20100101 Firefox/91.0",
  "requestParameters": {
    "maxResults": 20,
    "sort": {
      "sortBy": "RELEVANT",
      "sortOrder": "DESCENDING"
    }
  },
  "requestContext": {
    "integrationId": "HIDDEN_DUE_TO_SECURITY_REASONS"
  }
},
"responseElements": null,
"requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
"eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789010",
"eventCategory": "Management",
```

```
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "discovery.marketplace.us-east-1.amazonaws.com"
},
"sessionCredentialFromConsole": "true"
}
```

Notifications for the AWS Marketplace API

The following sections provide information about notifications.

Topics

- [Amazon EventBridge notifications for AWS Marketplace Catalog API events](#)

Amazon EventBridge notifications for AWS Marketplace Catalog API events

AWS Marketplace is integrated with Amazon EventBridge, formerly called Amazon CloudWatch Events. EventBridge is an event bus service that you can use to connect your applications with data from a variety of sources.

For information on how sellers, channel partners, and private marketplace administrators can receive ChangeSet status events using EventBridge, see [Events for change sets](#) in the *AWS Marketplace Seller Guide*.

Using the AWS Marketplace API as a seller

The following sections provide information on how to use the API as a seller.

Topics

- [Work with seller products](#)
- [Work with offers using the AWS Marketplace APIs](#)

Work with seller products

You can use the AWS Marketplace Catalog API to automate tasks for working with seller products. This includes the ability to create, update, view, list, and sort products. This enables you to automate product management. For example, you can provide self-service publishing capabilities on the AWS Marketplace Management Portal.

A *product* is a unit or resource that you intend to sell in AWS Marketplace, often referred to as a base product. Buyers can't consume a base product until you add product information, deployment attributes, and billing information.

A *product* describes the product information, software deployment attributes, and billing mechanism of the listing that you intend to sell. The *product* must be paired with an *offer* to become a transactable unit that you can sell and buyers can use in AWS Marketplace.

You can also use the AWS Marketplace Catalog API to:

- [Work with private offers using the AWS Marketplace APIs](#)
- [Work with resale authorizations using the AWS Marketplace APIs](#)
- [Work with channel partner private offers using the AWS Marketplace APIs](#)

Each product type has a different product entity. An *entity* can be a product or an offer on AWS Marketplace. The following product types and entities are supported:

Product type	Entity
Amazon Machine Image (AMI) products	AmiProduct@1.0

Product type	Entity
Container products	ContainerProduct@1.0
Software as a service (SaaS) products	SaaSProduct@1.0

Note

Single-AMI with AWS CloudFormation product types, AWS Data Exchange data products, and professional services products are not supported.

This chapter assumes that you have access to the API and have completed any seller prerequisites, as described in the [Access control for the AWS Marketplace Catalog API](#) topic.

See the following resources:

- To understand the basics of using the AWS Marketplace Catalog API, see [Welcome to the AWS Marketplace API Reference](#).
- For end-to-end labs with working code examples, see [Manage products with API](#) in the *AWS Marketplace seller workshop*.
- For code examples of API requests, see [Python](#) and [Java](#) examples in *AWS Samples* on GitHub.

The following topics describe how to use the Catalog API to perform actions on your single-AMI products, container-based products, or SaaS products.

Topics

- [Create a product](#)
- [Update product details](#)
- [Add pricing dimensions](#)
- [Update pricing dimensions](#)
- [Restrict pricing dimensions](#)
- [Update targeting configuration](#)
- [Update product visibility](#)
- [Publish a product](#)

- [Find your product ID](#)
- [Change set status and errors](#)
- [Work with single AMI products using the AWS Marketplace APIs](#)
- [Work with EC2 Image Builder component products](#)
- [Work with container-based products using the AWS Marketplace APIs](#)
- [Work with SaaS products using the AWS Marketplace APIs](#)

Create a product

Note

This change type is only needed when you intend to create a brand new product entity in the AWS Marketplace catalog. It is not needed when updating existing products.

You can use the Catalog API to create an AMI, container, or SaaS product document with identifiers (product code and product ID) in AWS Marketplace.

You create a product in Draft state by calling the StartChangeSet API operation with the CreateProduct change type.

If your request is processed successfully, then AWS Marketplace Catalog API generates a product in Draft state for you. This is an incomplete product and isn't visible to buyers in AWS Marketplace.

You then use Update change types to complete the create product process: [UpdateInformation](#), [UpdateDimensions](#), [UpdateTargeting](#), and [UpdateVisibility](#).

After the product is completed, you can use the [ReleaseProduct](#) change type to complete the product creation process, and then release the offer. This process validates the entire product and moves the product to the Limited state.

Note

For more information about creating a product using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- [Create your single-AMI product](#)

You cannot update the AMI for the version. If you need to update the AMI, create a new version instead.

- [Creating a container product](#)
- [Creating a SaaS product](#)

If you use the AWS Marketplace Management Portal to create a product, the product will be in the Staging state.

To create a product in Draft state, call the StartChangeSet API operation with the CreateProduct change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "SaaSProduct@1.0" // choose from ["AmiProduct@1.0",
"ContainerProduct@1.0", "SaaSProduct@1.0"]
      },
      "DetailsDocument": {
        "ProductTitle": "Test product title set in CreateProduct"
      }
    }
  ]
}
```

Provide information for the fields to add the CreateProduct change type. This change type can take in ProductTitle attribute, subject to the same restrictions as that sent into UpdateInformation change type.

- Entity (object) (required) – The named type of object being created.

- **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`, `ContainerProduct@1.0`, or `SaaSProduct@1.0`. For more information, see [Identifier](#).
- **DetailsDocument (object) (required)** – It may be empty.
- **ProductTitle (optional)** – The title for your product, max length is 72 characters. Note that you can also later set or update the product title via the `UpdateInformation` change type.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

When the request is complete (if the `Status` is `SUCCEEDED`), a new `ProductId` is generated.

Synchronous Validations

The following schema validations are specific to `CreateProduct` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
<code>ProductTitle</code> (string)	Max length: 72	400

Asynchronous Errors

The following errors are specific to `CreateProduct` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For

more details about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_INPUT	Inappropriate content '{InappropriateContent}' found in ProductTitle field. Provide ProductTitle with no inappropriate content.

Update product details

If you already have a product in AWS Marketplace, you can use the Catalog API to update the product details for an AMI, container, or SaaS product.

Note

For more information about updating the product details using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- AMI-based product: [Update product information](#)
- Container-based product: [Creating or updating product information for your container product](#)
- SaaS-based product: [Update product information](#)

To update product details, call the `StartChangeSet` API operation with the `UpdateInformation` change type and the details that you want to change, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
```

```
{
  "ChangeType": "UpdateInformation",
  "Entity":
  {
    "Identifier": "prod-example12345",
    "Type": "AmiProduct@1.0"
  },
  "DetailsDocument":
  {
    "ProductTitle": "My Product Title",
    "ShortDescription": "My product short description.",
    "LongDescription": "My product longer description.",
    "Sku": "123example456",
    "LogoUrl": "https://awsmp-
logos.s3.amazonaws.com/ca60b754fe05a24257176cdbf31c4e0d",
    "VideoUrls":
    [
      "https://example.com/my-video"
    ],
    "Highlights":
    [
      "123example45"
    ],
    "AdditionalResources":
    [
      {
        "Text": "123example456",
        "Url": "https://example.com/some-link"
      }
    ],
    "SupportDescription": "Need help? Contact our experts at support@example.com \n
\nYour purchase includes 24x7 support.",
    "Categories":
    [
      "Operating Systems",
      "Network Infrastructure",
      "Application Development"
    ],
    "SearchKeywords":
    [
      "123example456"
    ]
  }
}
```

```
]
}
```

Provide information for the fields to add the `UpdateInformation` change type:

- **Entity (object) (required)** – The named type of entity being created.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`, `ContainerProduct@1.0`, or `SaaSProduct@1.0`.
- **DetailsDocument (object) (required)** – The details of the request including the information that you want to update for your product. Each field is optional, but you must include at least one change to update.
 - **ProductTitle (string)** – The name of the product to be displayed to buyers.
 - **ShortDescription (string)** – The description of key aspects of the product to be displayed to buyers. This is usually 2–3 sentences.
 - **LongDescription (string)** – The longer description of your product to be displayed to buyers. This is usually 1–3 paragraphs.
 - **Skus (string or null)** – The free-form string that you define as a reference for your own use. Use `null` to unset this field.
 - **ImageUrl (string)** – The URL to an image in a publicly accessible Amazon Simple Storage Service (Amazon S3) bucket. For more information, see [Company and product logo requirements](#).
 - **VideoUrls (array of strings)** – The list of URLs to publicly available, externally hosted videos to be provided as a reference to buyers in your product information.

 **Note**

Currently, AWS Marketplace supports one URL in the array.

- **Highlights (array of strings)** – The list of short callouts for key product features.
- **AdditionalResources (array of structures)** – The list of references to additional resources to learn about your product. Each reference is made up of a text name and a URL:
 - **Text (string)** – The name or title of the resource.
 - **Url (string)** – The URL to a resource that might be helpful for a buyer to understand your

- **SupportDescription** (string) – The details about your support offering for your product.
- **Categories** (array of strings) – The list of AWS Marketplace defined product categories that describe your product. For more information, see [Product categories](#) in the *AWS Marketplace Buyer Guide*.
- **SearchKeywords** (array of strings) – The list of keywords for your product to enhance the search experience. Seller name, product name, and product categories are automatically included in search keywords and don't need to be repeated here.

Note

When you are initially populating product information (metadata) for a Draft product, you will need to supply all of the following in the `DetailsDocument` object of `UpdateInformation` change type: `ProductTitle`, `ShortDescription`, `LongDescription`, `LogoUrl`, `Highlights`, `AdditionalResources`, `SupportDescription`, `Categories`, and `SearchKeywords`.

The `ProductTitle` can be omitted if it has already been provided during `CreateProduct` change type. However, when you are updating existing fields on the product, you can include only the attributes that need to be changed in the `DetailsDocument` object of the `UpdateInformation` change type.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

To check request status, use the AWS Marketplace Management Portal or call the [DescribeChangeSet](#) API.

Synchronous Validations

The following schema validations are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
String (general)	No control characters "\\x00-\\x08\\x0B-\\x1F"	400
ProductTitle (string)	Max length: 72 Required	400
ShortDescription (string)	Max length: 1000 Required	400
LongDescription (string)	Max length: 5000 Required	400
Skus (string)	Max length: 100 Optional	400
LogoUrl (string)	URL pattern: ^https://(www\\.)?[-a-zA-Z0-9@.]{1,256}\\.[a-zA-Z0-9()]{2,63}\\b([-a-zA-Z0-9@+./]*)	400
VideoUrls (array of strings)	URL pattern: https://(www\\.)?[-a-zA-Z0-9@.]{1,256}\\.[a-zA-Z0-9()]{2,63}\\b([-a-zA-Z0-9@_+./\\])	400

Input field	Validation rule	HTTP code
Highlights (array of strings)	Required: Min 1 - Max 3	400
AdditionalResources (array of structures)	Max length: 500 Optional	400
SupportDescription (string)	Max length: 2000 Required	400
Categories (array of strings)	Required: Min 1 - Max 3 Required	400
SearchKeywords (array of strings)	Required: Min 1 - Max 3 Required	400

Asynchronous Errors

The following errors are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
MISSING_DATA	No data provided to perform an update. Provide data for at least 1 field of the product.
INVALID_INPUT	Provide LogoUrl.
INVALID_INPUT	Provide ProductTitle.
INVALID_INPUT	Provide ShortDescription.
INVALID_INPUT	Provide LongDescription.

Error code	Error message
INVALID_INPUT	Provide SupportDescription.
INVALID_INPUT	Provide at least one search keyword.
INVALID_INPUT	Provide at least one highlight.
INVALID_INPUT	Provide between 1 and 3 product categories.
INVALID_INPUT	Inappropriate content '{InappropriateContent}' found in ProductTitle field. Provide ProductTitle with no inappropriate content.
INVALID_INPUT	Inappropriate content '{InappropriateContent}' found in ShortDescription field. Provide ShortDescription with no inappropriate content.
INVALID_INPUT	Inappropriate content '{InappropriateContent}' found in LongDescription field. Provide LongDescription with no inappropriate content.
INVALID_INPUT	Inappropriate content '{InappropriateContent}' found in SupportDescription field. Provide SupportDescription with no inappropriate content.
INVALID_INPUT	Invalid ProductTitle field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid ShortDescription field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid LongDescription field. Remove spaces before trademark symbol.


Error code	Error message
INVALID_INPUT	Invalid SupportDescription field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid ProductTitle field. Remove unsupported characters [UnsupportedCharacters].
INVALID_INPUT	Invalid ShortDescription field. Remove unsupported characters [UnsupportedCharacters].
INVALID_INPUT	Invalid LongDescription field. Remove unsupported characters [UnsupportedCharacters].
INVALID_INPUT	Invalid SupportDescription field. Remove unsupported characters [UnsupportedCharacters].
INVALID_INPUT	Search keywords must be no more than 250 combined characters.
INVALID_INPUT	The input for this change type could not be read. Submit a properly formatted input.
INVALID_ADDITIONAL_RESOURCES	Invalid URLs in AdditionalResources: [InvalidAdditionalResourcesUrls] Provide valid URLs.
INVALID_CATEGORY_NAMES	Provide valid category names supported by AWS Marketplace.
InvalidImageProperties	Validation errors found: The file is not image type. Supported image types: [png jpg gif].
EXPLICIT_CONTENT	Explicit content: '{ExplicitContent}' detected. Provide media with no explicit content.

Error code	Error message
INVALID_MEDIA	Invalid URL: {MediaUrl} Provide a new URL for media stored in S3.
INVALID_MEDIA	Invalid URL: {MediaUrl} Provide a valid URL that does not exceed 2048 characters.
INVALID_MEDIA	Location provided not accessible: {MediaUrl} Provide an accessible URL for media stored in S3.
INVALID_MEDIA	There was an issue copying the media from S3. Image size exceeds 5 MB. Provide an image that is under 5 MB.
INVALID_MEDIA	There was an issue copying the media from S3. Video size exceeds 100 MB. Provide a video that is under 100 MB.
INVALID_MEDIA	Malware detected in media. Please resubmit media without malware.
INVALID_CHECKSUM	Invalid Sha256Base4 value provided: {ChecksumValue} Please provide a valid Sha256Base64 value.
TOO_MANY_MEDIA	Provide no more than 15 media items.
DUPLICATE_MEDIA	Duplicate media is not allowed for a product. Please provide media with no duplicates.

Add pricing dimensions

You can use the [the section called "AWS Marketplace Catalog API"](#) to add billable pricing dimensions that enable you to charge users for AMI, container, or SaaS products.

A *pricing dimension* is a unit of measure that sellers define for charging buyers. Sellers must set up this information to bill buyers for using the product, whether it's a usage-based or contract-based pricing model. The type of dimension depends on the product's pricing model.

 **Note**

New pricing dimension have the following impacts on SaaS buyers:

- For buyers with agreements created from public offers, you can report consumption on the new dimensions even though they did not exist in the offer when the agreement was created.
- For buyers with agreements created from private offers, you can't report consumption on the new dimensions because they did not exist in the private offer when the agreement was created. Calls to the [BatchMeterUsage](#) API will succeed, but the buyer will not be billed, so you must keep track of which buyer can and cannot be billed for any new dimensions. You can also use the [GetAgreementTerms](#) API to see the dimensions included in each buyer agreement.

To report consumption on a new dimension and bill the buyer, you must [extend a replacement offer](#) that includes the dimension, and the buyer must accept the offer.

For more information about product pricing, see the following topics in the *AWS Marketplace Seller Guide*:

- [AMI product pricing](#)
- [Container products pricing](#)
- [SaaS product pricing](#)

For more information about adding pricing dimensions using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- AMI-based product: [Update pricing](#)
- Container-based product: [Adding a pricing dimension](#)
- SaaS-based product: [Add pricing dimensions](#)

To add pricing dimensions, call the `StartChangeSet` API with the `AddDimensions` change type, as shown in the following example.

Note

After submitting the first `AddDimensions` change type with dimensions specifying a type of pricing model—usage, contract, or contract with consumption—you must work with the AWS Marketplace Seller Operations team. They help you add a dimension with types that are outside of the original pricing model.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "AddDimensions",
      "DetailsDocument":
      [
        {
          "Description": "Description of the dimension",
          "Key": "UniqueApiKey",
          "Unit": "HostHrs",
          "Name": "First Dimension",
          "Types":
          [
            "ExternallyMetered"
          ]
        }
      ],
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "SaaSProduct@1.0"
      }
    }
  ]
}
```

}

Provide the following fields for the AddDimensions change type.

- **DetailsDocument** (array of objects) (required) – Details of the request.
 - **Description** (string) (required) – Full details of the dimension that will be the long description on the buyer’s viewing page.
 - **Key** (string) (required) – Enter in the facet that will be used for defining the rates in the offer. Also, enter the dimensions published to the AWS Marketplace Metering Service (MMS) if the dimension can’t be metered externally. After the dimension is created, this can’t be changed.
 - **Units** (string) (required) – The unit type for the dimension. Possible units are Users, Hosts, GB, MB, TB, Gbps, Mbps, Requests, Units, UserHrs, UnitHrs, Units, HostHrs, TierHrs, and TaskHrs.
 - **Name** (string) (required) – The display name for the dimension on the website and customer's bill.
 - **Types** (array of strings) (required) (also known as **Tags**) – These indicate whether the dimension covers metering, entitlement, or support for external metering. This is not changeable after the dimension is created.
 - **Metered** – Indicates that Commerce Platform usage types should be created to allow metering to occur for this dimension.
 - **ExternallyMetered** – Indicates that AWS Marketplace Metering Service (MMS) dimensions should be created during publishing to allow sellers to meter through the AWS SDK.
 - **Entitled** – Indicates that entitlements can be granted for the dimension during the product or offer publishing.

The following table lists the supported combinations of pricing dimensions and products.

Pricing dimension type	Product types
[Metered]	AMI
[Metered, ExternallyMetered]	SaaS, AMI/Flexible Consumption Pricing (FCP) When ExternallyMetered appears, Metered is masked/inferred.

Pricing dimension type	Product types
[Entitled]	<p>SaaS Contracts, ProServ Products</p> <p>The Entitled tag grants rights to use a software/service start and end dates for the usage. Also, to grant rights to have usage discount for AMI annual products. Each entitlement is identified by a Dimension Key in AWS Marketplace Entitlement Service for creating or updating the entitlements.</p>
[Metered, ExternallyMetered, Entitled]	<p>Contracts with consumption pricing, where dimensions can be prepaid or metered, are a combination of both [ExternallyMetered] and [Entitled].</p>

- Entity (object) (required) – The named type of entity being created.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0 or SaaSProduct@1.0.

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to AddDimensions actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Description	Max length: 1000 Required	400
Key	Max length: 100 Pattern: [A-Za-z0-9_-.]+\$ Required	400
Dimension Units	Max length: 20 Required	400
Name	Max length: 500 Required	400
Type (tag)	Required: Min 1 - Max 3 Inputs: Entitled, Metered, ExternallyMetered Required	400

Asynchronous Errors

The following errors are specific to AddDimensions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For

more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
MISSING_DATA	No data provided to perform an update. Provide data for at least 1 dimension.
INVALID_DIMENSION	Provide no more than 24 dimensions.
INVALID_DIMENSION	Can't add duplicate dimensions.
INVALID_DIMENSION	Dimension can't be added in current state '%s'. States that support dimension updates are %s.
INVALID_DIMENSION	Can't add dimension. The field '%s' has duplicate values '%s' in other dimensions.
INVALID_DIMENSION	Provide non-empty fields (Key, Unit, Name, Types) for each dimension.
INVALID_TYPE	Remove invalid type '%s'. Valid types are ["Metered", "Entitled", "ExternallyMetered"].
INVALID_UNIT	Remove invalid Unit '%s'. Valid units are ["GB", "Gbps", "HostHrs", "Hosts", "MB", "Mbps", "Requests", "TaskHrs", "TB", "TierHrs", "UnitHrs", "Units", "UserHrs", "Users"].
INVALID_INPUT	Inappropriate content '%s' found in %s field. Provide %s with no inappropriate content.
INVALID_INPUT	Invalid '%s' field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid '%s' field. Remove unsupported characters %s.

Error code	Error message
INVALID_DIMENSION	Remove invalid dimension type combination %s. Allowed values are %s.
INVALID_DIMENSION	Remove invalid dimension key '%s' for Metered dimension.
INVALID_DIMENSION	Dimension named '%s' for productCode '%s' did not pass AWS Marketplace Metering Service validation %s.
INVALID_DIMENSION	Dimension named '%s' for productCode '%s' has no metering record present in Metering Service. The product has either never been launched for testing or is misconfigured and does not make the appropriate calls to the AWS Marketplace Metering Service.

Update pricing dimensions

You can use the Catalog API to update existing pricing dimensions of an AMI, container, or SaaS product in AWS Marketplace.

Each dimension is uniquely identified by the dimension key and dimension types to perform the update. Updating a dimension doesn't affect any active offer or customers that the original dimension had created.

Note

For more information about updating pricing dimensions using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- AMI-based product: [Update pricing](#)
- Container-based product: [Updating dimension information](#)
- SaaS-based product: [Update pricing dimensions](#)

To update pricing dimensions, call the `StartChangeSet` API operation with the `UpdateDimensions` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateDimensions",
      "DetailsDocument":
      [
        {
          "Key": "UniqueApiKey",
          "Types":
          [
            "ExternallyMetered"
          ],
          "Name": "First Dimension",
          "Description": "Description of the dimension"
        }
      ],
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "SaaSProduct@1.0"
      }
    }
  ]
}
```

Use the following fields with the `UpdateDimensions` change type:

- `DetailsDocument` (array of objects) (required) – Details of the request.
 - `Key` (string) (required) – Provide key of existing dimension from the product to change description and name on. For `UpdateDimension`, this field is only for identifying the dimension to be changed.

- **Types** (array of strings) (required) (also known as **Tags**) – These indicate whether the dimension covers metering, entitlement, or support for external metering. This is not changeable after the dimension is created.
- **Metered** – Indicates that Commerce Platform usage types should be created to allow metering to occur for this dimension.
- **ExternallyMetered** – Indicates that AWS Marketplace Metering Service (MMS) dimensions should be created during publishing to allow sellers to meter through the AWS SDK.
- **Entitled** – Indicates that entitlements can be granted for the dimension during product/offer publishing.

Valid Pricing Dimension Types Combinations

Pricing Dimension Type	Product
[Metered]	AMI
[ExternallyMetered]	SaaS, AMI/Flexible Consumption Pricing (FCP) When ExternallyMetered appears, Metered is masked/inferred.
[Entitled]	SaaS Contracts, ProServe Products The Entitled tag grants rights to use a software/service set start and end dates for the usage. Also, to grant rights to have usage discount for AMI annual products. Each entitlement is identified by a Dimension Key in AWS Marketplace Entitlement Service for creating or updating the entitlements.
[ExternallyMetered, Entitled]	Contracts with Consumption Pricing, where dimensions can be prepaid or metered are a combination of both [ExternallyMetered] and [Entitled].

- **Description** (string) (optional) – Full description of the dimension that will be the long description on the buyer's viewing page.
- **Name** (string) (optional) – Display name for the dimension on the website and customer's bill.
- **Entity** (object) (required) – The named type of entity being created.
 - **Identifier** (string) (required) – Your product ID. For more information, see [Identifier](#).
 - **Type** (string) (required) – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0` or `SaaSProduct@1.0`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateDimensions` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Description	Max length: 1000	400
	Required	

Input field	Validation rule	HTTP code
Key	Max length: 100 Pattern: [A-Za-z0-9_-.]+\$ Required	400
Name	Max length: 5 Required	400
Types (tag)	Required: Min 1 - Max 3 Inputs: Entitled, Metered, ExternallyMetered Required	422

Asynchronous Errors

The following errors are specific to UpdateDimensions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_INPUT	Invalid '%s' field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid '%s' field. Remove unsupported characters %s.
INVALID_DIMENSION	Provide non-empty fields (Key, Types, Name and/or Description) for each dimension.
INVALID_DIMENSION	Cannot update dimension. The field Name has duplicate values '%s' in other dimensions.

Error code	Error message
INVALID_DIMENSION	Cannot update same dimension with key '%s' and types '%s' multiple times in the same request.
INVALID_DIMENSION	Cannot restrict dimension. The dimension key '%s' with types '%s' does not exist.
INVALID_DIMENSION	Cannot update dimension. The dimension key '%s' is Metered.
INVALID_DIMENSION	Dimension cannot be updated for an already restricted dimension.

Restrict pricing dimensions

You can use the Catalog API to restrict existing pricing dimensions of an AMI or SaaS product in AWS Marketplace.

Each dimension is uniquely identified by the dimension key and dimension types to perform the update. Restricting a dimension doesn't affect any active offer or customers that the original dimension had created.

To restrict pricing dimensions, call the `StartChangeSet` API with the `RestrictDimensions` change type.

The following example shows how to restrict the `Entitled` dimension for a SaaS product.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "RestrictDimensions",
```

```
    "DetailsDocument":
    [
      {
        "Key": "UniqueApiKey",
        "Types": ["Entitled"]
      }
    ],
    "Entity":
    {
      "Identifier": "prod-example12345",
      "Type": "SaaSProduct@1.0"
    }
  }
]
```

Use the following fields with the `RestrictDimensions` change type:

- `DetailsDocument` (array of objects) (required) – Details of the request.
 - `Key` (string) (required) – Provide key of existing dimension from the product to change description and name on. For `RestrictDimensions`, this field is only for identifying the dimension to be changed.
 - `Types` (array of strings) (required) (also known as **Tags**) – These indicate whether the dimension covers metering, entitlement, or support for external metering. This is not changeable after the dimension is created.
 - `["ExternallyMetered"]`, `["Entitled"]` – You can only combine these types for SaaS Contract with Consumption pricing where dimensions can be prepaid or metered.
 - `["Metered"]` – For hourly-based pricing dimensions of AMI products. Indicates that Commerce Platform usage types should be created to allow metering to take place for this dimension.
 - `["ExternallyMetered"]` – For flexible consumption pricing dimensions (also known as custom metering) of AMI, container, and SaaS products. Indicates that AWS Marketplace Metering Service (MMS) dimensions should be created during publishing to allow sellers to meter through the AWS SDK.
 - `["Entitled"]` – For contract pricing dimensions of SaaS Contracts and professional services products. This tag grants rights to use a software or service, sets start and end dates for the usage, and grants usage-discount rights for AMI annual products. Each entitlement is identified by a Dimension Key in AWS Marketplace Entitlement Service for creating and

updating the entitlements. The key indicates that entitlements can be granted for the dimension during product and offer publishing.

- Entity (object) (required) – The named type of entity being created.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on your product's delivery method (product type): AmiProduct@1.0 or SaaSProduct@1.0.

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or by calling the [DescribeChangeSet](#) API.

Synchronous Validations

The following schema validations are specific to RestrictDimensions actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Description	Max length: 1000	400
	Required	
Key	Max length: 100	400

Input field	Validation rule	HTTP code
	Pattern: [A-Za-z0-9_-.]+\$ Required	
Name	Max length: 5 Required	400
Types (tag)	Required: Min 1 - Max 3 Inputs: Entitled, Metered, ExternallyMetered Required	400

Asynchronous Errors

The following errors are specific to RestrictDimensions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet while a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_INPUT	Invalid '%s' field. Remove spaces before trademark symbol.
INVALID_INPUT	Invalid '%s' field. Remove unsupported characters %s.
INVALID_DIMENSION	The dimension key '%s' with types '%s' was already restricted
INVALID_DIMENSION	Cannot restrict dimension. The dimension key '%s' with types '%s' does not exist
INVALID_DIMENSION	Cannot restrict duplicate dimensions.

Error code	Error message
INVALID_DIMENSION	All Entitled dimensions cannot be restricted. There must be at least one active Entitled dimension.
INVALID_DIMENSION	The dimension key '%s' with types '%s' is associated with another dimension of different types '%s'. Both dimensions of the same key must be restricted at the same time to be valid.

Update targeting configuration

You can use the Catalog API to add AWS account IDs that are allowed to view the AMI, container, or SaaS product in AWS Marketplace before it's moved to a `Public` state by calling the `UpdateVisibility` change type.

Note

For more information about adding AWS account IDs using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- AMI-based product: [Update the allowlist \(preview accounts\)](#)
- Container-based product: [Updating the allowlist of AWS account IDs](#)
- SaaS-based product: [Updating the allowlist of AWS account IDs](#)

To add AWS account IDs that are allowed to view the AMI, container, or SaaS product, call the `StartChangeSet` API operation with the `UpdateTargeting` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
```

```
"Catalog": "AWSMarketplace",
"ChangeSet":
[
  {
    "ChangeType": "UpdateTargeting",
    "Entity":
    {
      "Type": "SaaSProduct@1.0",
      "Identifier": "prod-example12345"
    },
    "DetailsDocument":
    {
      "PositiveTargeting":
      {
        "BuyerAccounts":
        [
          "1112223334444"
        ]
      }
    }
  }
]
```

Use the following the fields with the UpdateTargeting change type.

- Entity (object) (required) – The named type of entity being created.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0 or SaaSProduct@1.0.
- DetailsDocument (object) (required) – The details required to run the ChangeSet.
 - PositiveTargeting (object) (optional) – Positive targeting defines the criteria which any buyer's profile should fulfill in order to be allowed to access the offer. This field is optional, but at least one targeting option should be provided when this field is present.
 - BuyerAccounts (array of strings) (optional) – List as an option to allow targeting based on AWS accounts (also known as, Private Offer). If the intention is to not target the offer to an AWS account, this field should be omitted.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

To check request status, use the AWS Marketplace Management Portal or call the [DescribeChangeSet](#) API.

When the request is complete (if the Status is SUCCEEDED), a new ProductId is generated.

Synchronous Validations

The following schema validations are specific to UpdateTargeting actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Type of targeting	Valid current visibility states	BuyerAccounts (input)	Check
Positive	Public, Limited, or Draft	Array of 12-digit AWS account ID strings. Min size: 0. Max size: 5000.	The input must be different from the current document targeted accounts. Input must be in valid AWS accounts.

Asynchronous Errors

The following errors are specific to DescribeChangeSet actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is

processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_PRODUCT_VISIBILITY	Use an existing Public, Limited or Draft product.
INVALID_AWS_ACCOUNT_IDS	Provide valid AWS account IDs. AWS accounts not found: [x, y, z].

Update product visibility

You can use the Catalog API to update the visibility (also known as lifecycle state) of an AMI, container or SaaS product in AWS Marketplace.

Note

For more information updating product visibility using the AWS Marketplace Management Portal, see the following topics in the *AWS Marketplace Seller Guide*:

- AMI-based product: [Update product visibility](#)
- Container-based product: [Updating product visibility](#)
- SaaS-based product: [Update product visibility](#)

Allowed target lifecycle states are Limited, Public, or Restricted.

Limited

The product is complete and has successfully completed the `ReleaseProduct` `ChangeType`. Sellers can view details of the product in this state. The product is not public. However, sellers can target specific buyers to allow to preview the product.

Public

The product is visible in AWS Marketplace. Buyers can view and subscribe to the product.

Restricted

The product is no longer visible to the public and doesn't accept new subscribers. Existing subscribers can continue using this product until their subscription expires.

Note

The `UpdateVisibility` change type requires a manual review from the AWS Marketplace Seller Operations team, which results in a longer execution time. Use `UpdateVisibility` separately in its own change set.

To update your product's visibility, call the `StartChangeSet` API operation with the `UpdateVisibility` change type, as shown in the following example.

Request Syntax

For when `TargetVisibility` is `Public` or `Limited`.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateVisibility",
      "Entity":
      {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-example12345"
      },
      "DetailsDocument":
      {
        "TargetVisibility": "Public"
      }
    }
  ]
}
```

For when TargetVisibility is Restricted.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateVisibility",
      "Entity":
      {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-example12345"
      },
      "DetailsDocument":
      {
        "TargetVisibility": "Restricted",
        "ReplacementProductId": "prod-example54321"
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateVisibility change type.

- Entity (object) (required) – The named type of entity being created.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0, SaaSProduct@1.0, ContainerProduct@1.0. For more information, see [Identifier](#).
- DetailsDocument (object) (required) – The details required to run the ChangeSet.
 - TargetVisibility – The intended new visibility of the product.

Possible values: Public, Limited, and Restricted

- ReplacementProductId (string) (optional) – Replacement product ID for the product to be Restricted. Used to notify current subscribers about the product restriction.

Only accepts Restricted for TargetVisibility.

Synchronous Validations

The following schema validations are specific to UpdateVisibility actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Type of targeting	Valid current states	ReplacementProduct Id (input)	Validation checks
Public	Limited and Restricted	Not allowed	Valid current state
Limited	Public and Restricted	Not allowed	Valid current state
Restricted	Public and Limited	String (Optional)	ReplacementProduct Id must belong to an existing Limited or Public product.

After triggering this change type, it can take up to 37 days to complete. This includes the time the AWS Marketplace Seller Operations Team needs to review, audit, and approve. When restricting a product, you have 24 hours to change your mind, by calling `CancelChangeSet`, before the AWS Marketplace Seller Operations Team begins auditing. For more information, see [CancelChangeSet](#).

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

When the request is complete (if the Status is SUCCEEDED), a new ProductId is generated.

Asynchronous Errors

The following errors are specific to UpdateVisibility actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_PRODUCT_STATE	Use an existing Public, Limited, or Restricted product.
INVALID_TARGET_VISIBILITY	Provide a valid target visibility state: Public, Limited, or Restricted .
EMPTY_TARGET_VISIBILITY	Provide a valid target visibility state: Public, Limited, or Restricted .
INVALID_REPLACEMENT_PRODUCT_ID	Use an existing Public or Limited product as replacement.
INVALID_REPLACEMENT_PRODUCT_ID	Replacement product ID is only valid when restricting a product.
AUDIT_ERROR	Varies based on MCO manual review.
MISSING_SELLER_PROFILE_INFORMATION	Before you can update your product to Public, you must add a public profile to your seller account.

Publish a product

You can use the Catalog API to publish a Draft AMI, container, or SaaS product into Limited state in AWS Marketplace.

Note

For `AmiProduct@1.0` and `SaaSProduct@1.0`, the `ReleaseProduct` change type must be accompanied by `ReleaseOffer` change type on the corresponding draft public `Offer@1.0` entity created for this product.

To publish a product, call the `StartChangeSet` API operation with the `ReleaseProduct` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "ReleaseProduct",
      "Entity":
      {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-example12345"
      },
      "DetailsDocument": {}
    }
  ]
}
```

Provide information for the fields to add to the `ReleaseProduct` change type. This change type does not take any parameter payload.

- **Entity (object) (required)** – The named type of entity being created.

- **Identifier** (string) (required) – Your product ID. For more information, see [Identifier](#).
- **Type** (string) (required) – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`, `SaaSProduct@1.0`, `ContainerProduct@1.0`. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) - Must be an empty object. The change type `ReleaseProduct` doesn't accept any details.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

When the request is complete (if the Status is SUCCEEDED), a new `ProductId` is generated.

Asynchronous Errors

The following errors are specific to `ReleaseProduct` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
VALIDATION_FAILED	Provide Description information.

Error code	Error message
VALIDATION_FAILED	Provide Versions information.
VALIDATION_FAILED	Provide Dimensions information.
VALIDATION_FAILED	Provide Description PromotionalResources SupportInformation information.

Find your product ID

You must get the product ID for your product before you can modify it with AWS Marketplace Catalog API. There are two ways to find the product ID for server products:

- Open the AWS Marketplace Management Portal and sign in with your seller account. From the **Products** menu, select **Server products**, then choose the product you are interested in. The product ID is listed in the **Product Summary** section.
- Use the [ListEntities](#) action with the EntityType **AmiProduct** or **ContainerProduct**, **SaaSProduct**, or **DataProduct** to get a list of products, including their product IDs, via the Catalog API. `ListEntities` requires that you do not include the version of the entity type (for example, `AmiProduct@1.0`).

Note

The product ID is only available after your product has been published and is visible to at least yourself in AWS Marketplace. When you first create your product, it can take several days to be reviewed and fully created. During this time, it will not have a product ID available.

The following topics explain how to find a product by filtering on entity id, product title, last modified date, or visibility.

Topics

- [Find a product based on product title](#)
- [Find a product based on last modified date](#)

- [Find a product based on product visibility](#)
- [Find a product based on product title, last modified date, and product visibility](#)
- [Get additional details about a product](#)

Find a product based on product title

Request

```
POST /ListEntities HTTP/1.1
Content-Type: application/json

{
  "Catalog": "AWSMarketplace",
  "EntityType": "AmiProduct",
  "MaxResults": 10,
  "EntityTypeFilters": {
    "AmiProductFilters": {
      "ProductTitle": {
        "WildcardValue": "XYZ"
      }
    }
  }
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "EntitySummaryList": [
    {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example-abcd-1234",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12@1",
      "EntityType": "AmiProduct",
      "LastModifiedDate": "2018-02-27T13:45:22Z",
      "AmiProductSummary": {
        "ProductTitle": "ABC-XYZ-123",
        "Visibility": "Public"
      }
    }
  ]
}
```

```
    }
  }
],
"NextToken": ""
}
```

Find a product based on last modified date

Request

```
POST /ListEntities HTTP/1.1
Content-Type: application/json

{
  "Catalog": "AWSMarketplace",
  "EntityType": "AmiProduct",
  "MaxResults": 10,
  "EntityTypeFilters": {
    "AmiProductFilters": {
      "LastModifiedDate": {
        "DateRange": {
          "BeforeValue": "2018-03-27T13:45:22Z",
          "AfterValue": "2018-01-27T13:45:22Z"
        }
      }
    }
  }
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "EntitySummaryList": [
    {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example-abcd-1234",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12@1",
      "EntityType": "AmiProduct",
      "LastModifiedDate": "2018-02-27T13:45:22Z",
    }
  ]
}
```

```
    "AmiProductSummary": {
      "ProductTitle": "ABC-XYZ-123",
      "Visibility": "Public"
    }
  ],
  "NextToken": ""
}
```

Find a product based on product visibility

Request

```
POST /ListEntities HTTP/1.1
Content-Type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "EntityType": "AmiProduct",
  "MaxResults": 10,
  "EntityTypeFilters": {
    "AmiProductFilters": {
      "Visibility": {
        "ValueList": [
          "Public"
        ]
      }
    }
  }
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "EntitySummaryList": [
    {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example-abcd-1234",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12@1",
    }
  ]
}
```

```
    "EntityType": "AmiProduct",
    "LastModifiedDate": "2018-02-27T13:45:22Z",
    "AmiProductSummary": {
      "ProductTitle": "ABC-XYZ-123",
      "Visibility": "Public"
    }
  },
  "NextToken": ""
}
```

Find a product based on product title, last modified date, and product visibility

Request

```
POST /ListEntities HTTP/1.1
Content-Type: application/json

{
  "Catalog": "AWSMarketplace",
  "EntityType": "AmiProduct",
  "MaxResults": 10,
  "EntityTypeFilters": {
    "AmiProductFilters": {
      "LastModifiedDate": {
        "DateRange": {
          "BeforeValue": "2018-03-27T13:45:22Z",
          "AfterValue": "2018-01-27T13:45:22Z"
        }
      },
      "Visibility": {
        "ValueList": [
          "Public"
        ]
      },
      "ProductTitle": {
        "ValueList": [
          "ABC-XYZ-123"
        ]
      }
    }
  }
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "EntitySummaryList": [
    {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
AmiProduct/example-abcd-1234",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12@1",
      "EntityType": "AmiProduct",
      "LastModifiedDate": "2018-02-27T13:45:22Z",
      "AmiProductSummary": {
        "ProductTitle": "ABC-XYZ-123",
        "Visibility": "Public"
      }
    }
  ],
  "NextToken": ""
}
```

Get additional details about a product

You can get additional details about the product using the entity id with the DescribeEntity action.

Request

```
GET /DescribeEntity?catalog=AWSMarketplace&entityId=example-abcd-1234 HTTP/1.1
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "DetailsDocument": {
    "ProductTitle": "ABC-XYZ-123",
    "ShortDescription": "My product short description.",
    "LongDescription": "My product longer description."
  }
}
```



```
    "Sku": "123example456",
    "SupportDescription": "Need help? Contact our experts at support@example.com \n\nYour purchase includes 24x7 support.",
    "Categories": [
      "Operating Systems",
      "Network Infrastructure",
      "Application Development"
    ]
  }
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example-abcd-1234",
  "EntityId": "example1-abcd-1234-5ef6-7890abcdef12@1",
  "EntityType": "AmiProduct",
  "LastModifiedDate": "2018-02-27T13:45:22Z",
}
```

Change set status and errors

Making changes to seller products in the AWS Marketplace Catalog API involves creating change sets that describe the changes you want to make, and then using the `StartChangeSet` action to start the changes. The changes from the request can take minutes to hours or longer to complete, depending on the request. The response to this request looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed, including scanning the files and information to ensure that it meets the AWS Marketplace guidelines for products. Depending on the change requests, this process can take a few minutes to days. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the `DescribeChangeSet` action. For more information about change sets, see [Working with change sets](#).

To check the status of your request, use the `DescribeChangeSet` action.

```
POST /DescribeChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSetID": "example123456789012abcdef"
}
```

The result of this call looks like the following (in this case, for adding a new version to a container product).

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef",
  "ChangeSetName": "Submitted by 123456789012",
  "StartTime": "2020-10-27T22:21:26Z",
  "EndTime": "2020-10-27T22:32:19Z",
  "Status": "SUCCEEDED",
  "ChangeSet":
  [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity":
      {
        "Type": "ContainerProduct@1.0",
        "Identifier": "example-1234-abcd-56ef-abcdef12345678@4"
      },
      "Details": "{\n\"Version\": {\n\"VersionTitle\": \"1.1\", \"ReleaseNotes\": \"Minor bug fix\", \"DeliveryOptions\": [\n{\n\"DeliveryOptionTitle\": \"EKSDelivery\", \"Details\": {\n\"EcrDeliveryOptionDetails\": {\n\"ContainerImages\": [\n\"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"], \"DeploymentResources\": [\n{\n\"Name\": \"HelmDeploymentTemplate\", \"Url\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1\"}]\n}, \"CompatibleServices\": [\n\"EKS\"], \"Description\": \"Sample Description\", \"UsageInstructions\": \"helm pull 111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1\"}}], {\n\"DeliveryOptionTitle\": \"HelmChartDeliveryOption\", \"Details\": {\n\"HelmDeliveryOptionDetails\": {\n\"CompatibleServices\": [\n\"EKS\", \"EKS-Anywhere\"], \"ContainerImages\": [\n\"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"], \"HelmChartUri\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:helmchart1.1\", \"Description\": \"Helm chart description\", \"UsageInstructions\": \"Usage instructions\", \"QuickLaunchEnabled\": true, \"MarketplaceServiceAccountName\": \"Service account name\", \"ReleaseName\": \"Optional release name\", \"Namespace\": \"Optional Kubernetes namespace\", \"OverrideParameters\": [\n{\n\"Key\": \"HelmKeyName1\", \"DefaultValue\": \"${AWSMP_LICENSE_SECRET}\"}, {\n\"Metadata\": {\n\"Label\": \"AWS CloudFormation template field label\", \"Description\":
```

```

\ "AWS CloudFormation template field description\", \"Obfuscate\": false}}, {\"Key\":
\ "HelmKeyName2\", \"DefaultValue\": \"${AWSMP_SERVICE_ACCOUNT}\", \"Metadata\": {\"Label
\": \"AWS CloudFormation template field label\", \"Description\": \"AWS CloudFormation
template field description\", \"Obfuscate\": false}}]}]}]}]}\",
  \"DetailsDocument\":
  {
    \"Version\":
    {
      \"VersionTitle\": \"1.1\",
      \"ReleaseNotes\": \"Minor bug fix\"
    },
    \"DeliveryOptions\":
    [
      {
        \"DeliveryOptionTitle\": \"EKSDelivery\",
        \"Details\":
        {
          \"EcrDeliveryOptionDetails\":
          {
            \"ContainerImages\":
            [
              \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:1.1\"
            ],
            \"DeploymentResources\":
            [
              {
                \"Name\": \"HelmDeploymentTemplate\",
                \"Url\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame2:mychart1.1\"
              }
            ],
            \"CompatibleServices\":
            [
              \"EKS\"
            ],
            \"Description\": \"Sample Description\",
            \"UsageInstructions\": \"helm pull 111122223333.dkr.ecr.us-
east-1.amazonaws.com/sellername/reponame2:mychart1.1\"
          }
        }
      },
      {
        \"DeliveryOptionTitle\": \"HelmChartDeliveryOption\",

```

```

"Details":
{
  "HelmDeliveryOptionDetails":
  {
    "CompatibleServices":
    [
      "EKS",
      "EKS-Anywhere"
    ],
    "ContainerImages":
    [
      "111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:1.1"
    ],
    "HelmChartUri": "111122223333.dkr.ecr.us-east-1.amazonaws.com/
sellername/reponame1:helmchart1.1",
    "Description": "Helm chart description",
    "UsageInstructions": "Usage instructions",
    "QuickLaunchEnabled": true,
    "MarketplaceServiceAccountName": "Service account name",
    "ReleaseName": "Optional release name",
    "Namespace": "Optional Kubernetes namespace",
    "OverrideParameters":
    [
      {
        "Key": "HelmKeyName1",
        "DefaultValue": "${AWSMP_LICENSE_SECRET}",
        "Metadata":
        {
          "Label": "AWS CloudFormation template field label",
          "Description": "AWS CloudFormation template field description",
          "Obfuscate": false
        }
      },
      {
        "Key": "HelmKeyName2",
        "DefaultValue": "${AWSMP_SERVICE_ACCOUNT}",
        "Metadata":
        {
          "Label": "AWS CloudFormation template field label",
          "Description": "AWS CloudFormation template field description",
          "Obfuscate": false
        }
      }
    ]
  }
}

```

```

    ]
  }
}
]
},
"ErrorDetailList":
[]
}
]
}

```

The `Status` field shows the current status of the request, in this case, `SUCCEEDED`.

If there are failures, the result can include two types of errors. For most errors, the error message is included directly. However, errors found while scanning the product for security vulnerabilities instead include a URL to a file that lists all of the errors found, in the `ErrorMessage` field. Errors found while scanning have the `ErrorCode` `SCAN_ERROR`.

```

{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef",
  "ChangeSetName": "Submitted by 123456789012",
  "StartTime": "2020-10-27T22:21:26Z",
  "EndTime": "2020-10-27T22:32:19Z",
  "Status": "FAILED",
  "FailureDescription": "Change set preparation has failed. For details see
'ErrorDetailList'.",
  "ChangeSet":
  [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity":
      {
        "Type": "ContainerProduct@1.0",
        "Identifier": "example-1234-abcd-56ef-abcdef12345678@4"
      },
      "Details": "{\"Version\": {\"VersionTitle\": \"1.1\", \"ReleaseNotes\": \"Minor
bug fix\"}, \"DeliveryOptions\": [{\"DeliveryOptionTitle\": \"EKSDelivery\", \"Details
\": {\"EcrDeliveryOptionDetails\": {\"ContainerImages\": [\"111122223333.dkr.ecr.us-
east-1.amazonaws.com/sellername/reponame1:1.1\"], \"DeploymentResources\": [{\"Name\":

```

```

    \"HelmDeploymentTemplate\", \"Url\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/
sellername/reponame2:mychart1.1\"}], \"CompatibleServices\": [\"EKS\"], \"Description
\": \"Sample Description\", \"UsageInstructions\": \"helm pull 111122223333.dkr.ecr.us-
east-1.amazonaws.com/sellername/reponame2:mychart1.1\"}}], {\"DeliveryOptionTitle
\": \"HelmChartDeliveryOption\", \"Details\": {\"HelmDeliveryOptionDetails\":
{\"CompatibleServices\": [\"EKS\", \"EKS-Anywhere\"], \"ContainerImages\":
[\"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"],
\"HelmChartUri\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:helmchart1.1\", \"Description\": \"Helm chart description\",
\"UsageInstructions\": \"Usage instructions\", \"QuickLaunchEnabled\": true,
\"MarketplaceServiceAccountName\": \"Service account name\", \"ReleaseName\": \"Optional
release name\", \"Namespace\": \"Optional Kubernetes namespace\", \"OverrideParameters
\": [{\"Key\": \"HelmKeyName1\", \"DefaultValue\": \"${AWSMP_LICENSE_SECRET}\",
\"Metadata\": {\"Label\": \"AWS CloudFormation template field label\", \"Description\":
\"AWS CloudFormation template field description\", \"Obfuscate\": false}}, {\"Key\":
\"HelmKeyName2\", \"DefaultValue\": \"${AWSMP_SERVICE_ACCOUNT}\", \"Metadata\": {\"Label
\": \"AWS CloudFormation template field label\", \"Description\": \"AWS CloudFormation
template field description\", \"Obfuscate\": false}}]}]}]}]}\",
  \"DetailsDocument\":
  {
    \"Version\":
    {
      \"VersionTitle\": \"1.1\",
      \"ReleaseNotes\": \"Minor bug fix\"
    },
    \"DeliveryOptions\":
    [
      {
        \"DeliveryOptionTitle\": \"EKSDelivery\",
        \"Details\":
        {
          \"EcrDeliveryOptionDetails\":
          {
            \"ContainerImages\":
            [
              \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:1.1\"
            ],
            \"DeploymentResources\":
            [
              {
                \"Name\": \"HelmDeploymentTemplate\",
                \"Url\": \"111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame2:mychart1.1\"
              }
            ]
          }
        }
      }
    ]
  }
}

```

```

    }
  ],
  "CompatibleServices":
  [
    "EKS"
  ],
  "Description": "Sample Description",
  "UsageInstructions": "helm pull 111122223333.dkr.ecr.us-
east-1.amazonaws.com/sellername/reponame2:mychart1.1"
}
},
{
  "DeliveryOptionTitle": "HelmChartDeliveryOption",
  "Details":
  {
    "HelmDeliveryOptionDetails":
    {
      "CompatibleServices":
      [
        "EKS",
        "EKS-Anywhere"
      ],
      "ContainerImages":
      [
        "111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:1.1"
      ],
      "HelmChartUri": "111122223333.dkr.ecr.us-east-1.amazonaws.com/
sellername/reponame1:helmchart1.1",
      "Description": "Helm chart description",
      "UsageInstructions": "Usage instructions",
      "QuickLaunchEnabled": true,
      "MarketplaceServiceAccountName": "Service account name",
      "ReleaseName": "Optional release name",
      "Namespace": "Optional Kubernetes namespace",
      "OverrideParameters":
      [
        {
          "Key": "HelmKeyName1",
          "DefaultValue": "${AWSMP_LICENSE_SECRET}",
          "Metadata":
          {
            "Label": "AWS CloudFormation template field label",

```

```

        "Description": "AWS CloudFormation template field description",
        "Obfuscate": false
    }
},
{
    "Key": "HelmKeyName2",
    "DefaultValue": "${AWSMP_SERVICE_ACCOUNT}",
    "Metadata":
    {
        "Label": "AWS CloudFormation template field label",
        "Description": "AWS CloudFormation template field description",
        "Obfuscate": false
    }
}
]
}
}
]
},
"ErrorDetailList":
[
    {
        "ErrorCode": "DUPLICATE_VERSION_TITLE",
        "ErrorMessage": "The version title must be different from any other version
titles of this product."
    },
    {
        "ErrorCode": "SCAN_ERROR",
        "ErrorMessage": "https://123sample456.cloudfront.net/example-1234-abcd-5678-
abcdef12345678/1234abcdef567890"
    }
]
}
]
}
}

```

In this example, there is one error directly reported (DUPLICATE_VERSION_TITLE). The other error has a file with error messages (a single SCAN_ERROR can have multiple found errors in the file that is linked).

Note

The link returned in the `ErrorMessage` is valid for 60 days.

Work with single AMI products using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with single Amazon Machine Image (AMI)-based products.

For information about creating a single AMI-based product using the Catalog API, see [Create a product](#).

The following topics describe how to use the Catalog API to perform actions on your single AMI-based products:

Topics

- [Add a new version](#)
- [Update version information](#)
- [Restrict a version](#)
- [Update future AWS Region support](#)
- [Add a supported AWS Region](#)
- [Restrict an AWS Region](#)
- [Add a new instance type](#)
- [Restrict an instance type](#)

Note

You can also create a single AMI-based product using the AWS Marketplace Management Portal. For more information, see [Single-AMI products](#) in the *AWS Marketplace Seller Guide*. For a walk-through showing how to automate updating your AMI-based product, you can also refer to the video, [Automating updates to your product listings in AWS Marketplace with Catalog API](#) (5:08).

Add a new version

You can use the Catalog API to add a new version to an existing AMI-based product in AWS Marketplace. For more information about adding new AMI versions to your product using the AWS Marketplace Management Portal, see [Adding a new version](#) in the *AWS Marketplace Seller Guide*.

To add a new version, call the `StartChangeSet` API operation with the `AddDeliveryOptions` change type for single-AMI products, as shown in the following example. To test your API call without actually creating a new version, set the `Intent` parameter to `VALIDATE`. For more information, see [Intent](#).

Note

For single-AMI products, a version is made up of a single delivery option, which is the AMI that you are making available. In the Catalog API, working with delivery options for single-AMI products modifies versions of your product.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity":
      {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12@1",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument":
      {
        "Version":
        {
          "VersionTitle": "My new title",
          "ReleaseNotes": "My new Release notes"
        },
        "DeliveryOptions":
```

```

    [
      {
        "Details":
        {
          "AmiDeliveryOptionDetails":
          {
            "AmiSource":
            {
              "AmiId": "ami-1234567890abcdef",
              "AccessRoleArn": "arn:aws:iam::12345678901:role/
HVM gp2"
              "UserName": "ec2-user",
              "OperatingSystemName": "AMAZONLINUX",
              "OperatingSystemVersion": "Amazon Linux 2 AMI 2.0.20210126.0 x86_64"
            },
            "UsageInstructions": "Easy to use AMI",
            "RecommendedInstanceType": "m4.xlarge",
            "SecurityGroups":
            [
              {
                "IpProtocol": "tcp",
                "FromPort": 443,
                "ToPort": 443,
                "IpRanges":
                [
                  "0.0.0.0/0"
                ]
              }
            ]
          }
        }
      }
    ],
    "Intent": "APPLY"
  }

```

The following is information about the input fields you provide for adding the AddDeliveryOptions change type. For more information about these fields, see [Adding a new version](#) in the AWS Marketplace Seller Guide.

- **Entity (object) (required)** – Your AMI-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`.
- **DetailsDocument (object) (required)** – Details of the request. It includes all the information about the new version of your AMI-based product.
- **Version (object)** – Details about the software version you are adding. Made up of a title and release notes.
 - **VersionTitle (string)** – Unique name of the version. Displayed to end users in product details page and configuration pages for the product in AWS Marketplace.
 - **ReleaseNotes (string)** – Notes for buyers to tell them about changes from one version to the next.
- **DeliveryOptions (array)** – List of `DeliveryOption` objects, including the details of each:
 - **Details (object)** – Holds the details of an AMI delivery option. Note that this nested details object does *not* need to be double-escaped.
 - **AmiDeliveryOptionDetails (object)** – The details of one AMI delivery option.
 - **AmiSource (object)** – Details about the AMI to be used for the added version.
 - **AmiId (string)** – ID for the source AMI, located in the AWS Region where the API is being called (currently must always be US East (N. Virginia) because that is the only Region where the Catalog API is available). Must belong to the caller account.
 - **AccessRoleArn (string)** – IAM role Amazon Resource Name (ARN) used by AWS Marketplace to access the provided AMI. For details about creating and using this ARN, see [Giving AWS Marketplace access to your AMI](#) in the *AWS Marketplace Seller Guide*.
 - **UserName (string)** – Login user name to access the operating system (OS) in the AMI. Typically `ec2-user` for Linux AMIs or `Administrator` for Windows.
 - **ScanningPort (integer)** – SSH or RDP port used to access the OS. Used for scanning the provided AMI for security vulnerabilities. Defaults to 22.
 - **OperatingSystemName (string)** – Name of the operating system displayed to buyers.
 - **OperatingSystemVersion (string)** – Operating system version string displayed to buyers.
 - **UsageInstructions (string)** – Instructions for using the AMI, or a link to more information about the AMI.

- `AccessEndpointUrl` (object) – Used to create a path to access the AMI after it is used.
- `Port` (string) – The port number used to access the service running on the AMI.
- `Protocol` (string) – The protocol (`http` or `https`) used to access the service running on the AMI.
- `RelativePath` (string) – The path from the web root to access the service running on the AMI (for example `/index.html`).
- `RecommendedInstanceType` (string) – The instance type that is recommended to run the service with the AMI and is the default for 1-click installs of your service. For a list of instance types, see [Instance types](#) in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*.
- `SecurityGroups` (array of objects) – A list of objects representing ingress rules for the automatically created groups for the version.
 - `FromPort` (integer) – The source port.
 - `IpProtocol` (string) – The protocol to use (`tcp` or `udp`).
 - `IpRanges` (array of strings) – IP ranges to allow, in CIDR format (in the form `xxx.xxx.xxx.xxx/nn`, for example, `192.0.2.0/24`).
 - `ToPort` (integer) – The destination port.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

For more information about errors in seller product change sets, see [Change set status and errors](#).

When the request is complete, the version is added, and any existing subscribers will receive an email message telling them about the new version. For more information about the process of adding a new version, see [Adding a new version](#) in the *AWS Marketplace Seller Guide*.

Asynchronous Errors

The following errors are specific to `AddDeliveryOptions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_PRODUCT	Use an existing limited or public product.
DUPLICATE_VERSION_TITLE	The version title must be different from any other version titles of this product.
INVALID_VERSION_TITLE	Remove spaces before the trademark symbol.
INVALID_VERSION_TITLE	Remove unsupported characters: [x, y, z]
INVALID_VERSION_TITLE	Remove spaces from the beginning of the version title.
INVALID_VERSION_TITLE	Provide version title with fewer than [x] characters.
INVALID_RELEASE_NOTES	Remove spaces before the trademark symbol.
INVALID_RELEASE_NOTES	Remove unsupported characters: [x, y, z]
INVALID_RELEASE_NOTES	Remove spaces from the beginning of release notes.
INVALID_RELEASE_NOTES	Provide release notes with fewer than (x) characters.
INVALID_USAGE_INSTRUCTIONS	Remove spaces before the trademark symbol.

Error code	Error message
INVALID_USAGE_INSTRUCTIONS	Remove unsupported characters: [x, y, z]
INVALID_USAGE_INSTRUCTIONS	Remove spaces from the beginning of release notes.
INVALID_USAGE_INSTRUCTIONS	Provide usage instructions with fewer than (x) characters.
RECOMMENDED_INSTANCE_TYPE_NOT_AVAILABLE	Provide an existing, available instance type.
INVALID_RECOMMENDED_INSTANCE_TYPE	Provide a valid instance type.
INVALID_SECURITY_GROUP	Security group ports must be between 1 and [max].
INVALID_SECURITY_GROUP	Provide a value for CIDR IP ranges.
INVALID_SECURITY_GROUP	Provide security group start port that is not greater than end port.
INVALID_SECURITY_GROUP_PROTOCOL	Security group protocol must either be 'tcp' or 'udp'.
INVALID_CIDR_IP	Provide standard CIDR IP range in form '0.0.0.0/0'.
INVALID_ACCESS_ENDPOINT_PORT	Provide endpoint port less than [x].
INVALID_ACCESS_ENDPOINT_PORT	Provide endpoint port between 1 and [max].
INVALID_ACCESS_ENDPOINT_PORT	Provide endpoint port.
INVALID_ACCESS_ENDPOINT_RELATIVE_PATH	Remove spaces in the relative path.
INVALID_ACCESS_ENDPOINT_RELATIVE_PATH	Remove preceding '/' from relative path.
INCOMPATIBLE_OPERATING_SYSTEM	Provide operating system name and version that is compatible with instance types: [x]

Error code	Error message
INCOMPATIBLE_OPERATING_SYSTEM_NAME	Provide name with fewer than (x) characters.
INCOMPATIBLE_OPERATING_SYSTEM_NAME	Provide operating system name that is supported.
INCOMPATIBLE_OPERATING_SYSTEM_VERSION	Provide version with fewer than (x) characters.
INVALID_SCANNING_PORT	Provide scanning port between 1 and [max].
INVALID_AMI_ID	Provide valid AMI ID.
EXISTING_AMI_PRODUCT_CODE	Remove product code attached to image X.
INVALID_AMI_ARCHITECTURE	Provide new AMI with architecture [x].
INVALID_AMI_VIRTUALIZATION_TYPE	Provide new AMI with virtualization type [x].
INVALID_AMI_VIRTUALIZATION_TYPE	Provide expected [z] volume on image [x].
INCOMPATIBLE_AMI	Provide new AMI as architecture [x] on [y] is not supported by following instance types: [z]
INCOMPATIBLE_AMI	Provide new AMI as virtualization type [x] on [y] is not supported by following instance types: [z]
INCOMPATIBLE_AMI	Enable ENA support for image x because following instance types require ENA support: [y]
ASSET_NOT_FOUND	Check if [ami-id] exists in us-east-1 Region of [account-id] AWS account and the AccessARN provided [ARN] has permissions to share this AMI with AWS Marketplace.

Error code	Error message
ASSET_ACCESS_EXCEPTION	Unable to copy AMI [x] into AWS Marketplace account.
SCAN_ERROR	Fix security vulnerability [y] on Image [x].

Update version information

You can use the Catalog API to update the details of an existing version of your AMI-based product in AWS Marketplace.

Note

For more information about updating version information using the AWS Marketplace Management Portal, see [Updating version information](#) in the *AWS Marketplace Seller Guide*.

You cannot update the AMI for the version. If you need to update the AMI, create a new version instead.

To add a new version, call the `StartChangeSet` API operation with the `UpdateDeliveryOptions` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity":
      {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12@1",
```

```

    "Type": "AmiProduct@1.0"
  },
  "DetailsDocument":
  {
    "Version":
    {
      "ReleaseNotes": "My new Release notes"
    },
    "DeliveryOptions":
    [
      {
        "Id": "example1-2222-cccc-2222-cccccccccccc",
        "Details":
        {
          "AmiDeliveryOptionDetails":
          {
            "UsageInstructions": "Easy to use AMI"
          }
        }
      }
    ]
  }
}
]
}

```

The following is information about the input fields you provide for adding the UpdateDeliveryOptions change type. For more information about these fields, see [Updating version information](#) in the AWS Marketplace Seller Guide.

- Entity (object) (required) – Your AMI-based product.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0.
- DetailsDocument (object) (required) – Details of the request. It includes any information about the version of your AMI-based product that you would like to update. The included fields are all optional, but you must include at least one field to update.
 - Version (object) – Details about the software version.
 - ReleaseNotes (string) – Notes for buyers to tell them about changes from one version to the next.

- `DeliveryOptions` (array of objects) – List of `DeliveryOption` objects, including the details of each:
 - `Id` (string) – Unique identifier for the `DeliveryOption` (you can get the unique identifier for the `DeliveryOption` by calling the `DescribeEntity` action on the product you are updating).
 - `Details` (object) – Holds the details of an AMI delivery option. Note that this nested details object does *not* need to be double-escaped.
 - `AmiDeliveryOptionDetails` (object) – The details of one AMI delivery option.
 - `UsageInstructions` (string) – Instructions for using the AMI, or a link to more information about the AMI.
 - `AccessEndpointUrl` (object) – Used to create a path to access the AMI after it is used.
 - `Port` (string) – The port number used to access the service running on the AMI.
 - `Protocol` (string) – The protocol (http or https) used to access the service running on the AMI.
 - `RelativePath` (string) – The path from the web root to access the service running on the AMI (for example `/index.html`).
 - `RecommendedInstanceType` (string) – The instance type that is recommended to run the service with the AMI and is the default for 1-click installs of your service.
 - `SecurityGroups` (array of objects) – A list of objects representing ingress rules for the automatically created groups for the version:
 - `FromPort` (integer) – The source port.
 - `IpProtocol` (string) – The protocol to use (tcp or idp).
 - `IpRanges` (array of strings) – IP ranges to allow, in CIDR format (in the form `xxx.xxx.xxx.xxx/nn`, for example, `192.0.2.0/24`).
 - `ToPort` (integer) – The destination port.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
```

```
"ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

Asynchronous Errors

The following errors are specific to UpdateDeliveryOptions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_PRODUCT	Use an existing limited or public product.
MISSING_DELIVERY_OPTION_IDS	Provide at least one delivery option ID.
INVALID_DELIVERY_OPTION_IDS	Provide delivery option IDs that can be found in the product. IDs not found: [x]
INVALID_DELIVERY_OPTIONS	Provide delivery option IDs that belong to the same version.

Restrict a version

You can use the Catalog API to restrict a version of your AMI-based product in AWS Marketplace. This prevents new buyers from being able to use that version. There must always be at least one unrestricted version of a product available, so you cannot restrict the last publicly available version for a product.

Note

For more information about restricting AMI versions in AWS Marketplace via the AWS Marketplace Management Portal, see [Restricting a version](#) in the *AWS Marketplace Seller Guide*.

To restrict a version, call the `StartChangeSet` API operation with the `RestrictDeliveryOptions` change type, as shown in the following example.

Note

All subscribers can use the current version regardless of the restriction status. AWS Marketplace guidelines require that you continue to offer support to existing buyers for 90 days after restricting the version. Your AMI will be marked as deprecated after the version is restricted. For more information, see [Deprecate an AMI](#) in the *Amazon Elastic Compute Cloud User Guide for Windows Instances*.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "RestrictDeliveryOptions",
      "Entity":
      {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12@1",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument":
      {
        "DeliveryOptionIds":
        [
          "example1-2222-cccc-2222-cccccccccccc"
        ]
      }
    }
  ]
}
```

```
    ]
  }
}
]
```

The following is information about the input fields you provide for adding the `RestrictDeliveryOptions` change type:

- **Entity (object) (required)** – Your AMI-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`.
- **DetailsDocument (object) (required)** – Details of the request. It includes IDs for the versions of your AMI-based product that you would like to restrict.
 - **DeliveryOptionIds (array of objects)** – List of `DeliveryOption` IDs for the versions that you want to restrict. You can get the unique identifier for the `DeliveryOption` by calling the `DescribeEntity` action on the version you are restricting.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Asynchronous Errors

The following errors are specific to `RestrictDeliveryOptions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_PRODUCT	Use an existing public product.
MISSING_DELIVERY_OPTION_IDS	Provide at least one delivery option ID.
INVALID_DELIVERY_OPTION_IDS	Provide delivery option IDs that can be found in the product. IDs not found: [x]
INVALID_DELIVERY_OPTION	Provide delivery option IDs that are in a public state. IDs not in public state: [x]
ALL_DELIVERY_OPTIONS_RESTRICTED	Provide fewer delivery options to restrict as at least one must remain in public state.

Update future AWS Region support

You can use the Catalog API to change future AWS Region support preferences for your AMI-based product in AWS Marketplace.

Note

For more information about changing future Region support using the AWS Marketplace Management Portal, see [Update support for future AWS Regions](#) in the *AWS Marketplace Seller Guide*.

Note

The `UpdateFutureRegionSupport` change type is only available on `AmiProduct@1.0`.

To change future AWS Region support preferences, call the StartChangeSet API operation with the UpdateFutureRegionSupport change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateFutureRegionSupport",
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument":
      {
        "FutureRegionSupport":
        {
          "SupportedRegions":
          [
            "All"
          ]
        }
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateFutureRegionSupport change type:

- Entity (object) (required) – Your AMI-based product.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0.
- DetailsDocument (object) (required) – The details required to execute the ChangeSet.
 - FutureRegionSupport – Object

- `SupportedRegions` – Single-element array of strings

Element supported values: one of ["All", "US", "None"]

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Asynchronous Errors

The following errors are specific to `UpdateFutureRegionSupport` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
TOO_MANY_REGIONS	Currently, only 1 value is supported for <code>FutureRegionSupport</code> : All, US, or None
INVALID_REGIONS	Requested Regions [a, b, c] are invalid or unavailable. Only supported values are [x, y, z].
INVALID_INPUT	<code>SupportedRegions</code> can't be empty.

Add a supported AWS Region

You can use the Catalog API to add new supported AWS Regions for your AMI-based product in AWS Marketplace.

Note

For more information about adding new supported Regions using the AWS Marketplace Management Portal, see [Add an AWS Region](#) in the *AWS Marketplace Seller Guide*.

Note

The AddRegions change type is only available on AmiProduct@1.0.

To add new supported Regions, call the StartChangeSet API operation with the AddRegions change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddRegions",
      "DetailsDocument": {
        "Regions": [
          "us-east-1",
          "ap-northeast-2"
        ]
      },
      "Entity": {
        "Identifier": "prod-123456@1",
        "Type": "AmiProduct@1.0"
      }
    }
  ]
}
```

```
}

```

Provide information for the fields to add the AddRegions change type.

- Entity (object) (required) – Your AMI-based product.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: AmiProduct@1.0.

- **Example**

<caption>DetailsDocument (object) (required) – The details required to execute the ChangeSet.</caption>

- Regions: Array of strings

Element supported values: Valid AWS Region code strings.

For example, ["us-east-1"].

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

- If the change set execution status is SUCCEEDED: A new Entity Identifier (or EntityId) is generated. You can use the [DescribeEntity](#) API operation on the product entity to check the result.

- If the change set execution status is `CLIENT_ERROR`: The `DescribeChangeSet` response gives the details of the error, as well as corresponding actions to take to fix the error.

Asynchronous Errors

The following errors are specific to `AddRegions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
<code>INVALID_REGIONS</code>	Requested regions [a, b, c] are invalid or unavailable. Only supported values are [x, y, z].
<code>INVALID_INPUT</code>	Regions can't be empty.

Restrict an AWS Region

You can use the Catalog API to restrict previously supported AWS Regions for your AMI-based product in AWS Marketplace.

Note

For more information about restricting previously supported Regions using the AWS Marketplace Management Portal, see [Restrict an AWS Region](#) in the *AWS Marketplace Seller Guide*.

Note

The `RestrictRegions` change type is only available on `AmiProduct@1.0`.

To restrict previously supported Regions, call the `StartChangeSet` API operation with the `RestrictRegions` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictRegions",
      "DetailsDocument": {
        "Regions": [
          "us-east-1",
          "ap-northeast-2"
        ]
      },
      "Entity": {
        "Identifier": "prod-123456@1",
        "Type": "AmiProduct@1.0"
      }
    }
  ]
}
```

Provide information for the fields to add the AddRegions change type.

- **Entity** (object) (required) – Your AMI-based product.
 - **Identifier** (string) (required) – Your product ID. For more information, see [Identifier](#).
 - **Type** (string) (required) – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`.
- **DetailsDocument** (object) (required) – The details required to execute the ChangeSet.
 - **Regions** – Array of strings

Element supported values: Valid AWS Region code strings, such as "us-east-1".

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

- If the change set execution status is SUCCEEDED – A new Entity Identifier (or EntityId) is generated. You can use DescribeEntity on the product entity to check the result. For more information, see [DescribeEntity](#).
- If the change set execution status is CLIENT_ERROR: The DescribeChangeSet response gives the details of the error, as well as corresponding actions to take to fix the error.

Asynchronous Errors

The following errors are specific to AddRegions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_REGIONS	Requested regions [a, b, c] are invalid or unavailable. Only supported values are [x, y, z].
INVALID_INPUT	Regions can't be empty.

Add a new instance type

You can use the Catalog API to add new instance types for your AMI-based product in AWS Marketplace.

Note

For more information about adding instance types using the AWS Marketplace Management Portal, see [Add an instance](#) in the *AWS Marketplace Seller Guide*.

AddInstanceTypes will add new instance types to existing products and newly created products when creating a product. The change type will update all versions in product document with a new instance type.

Note

The AddInstanceTypes change type is only available on AmiProduct@1.0.

When adding a restricted instance type, the instance type can be removed from the restricted list and added to the available instance type list. This gives sellers more control to change their product restriction. The instance type list is interchangeable and not a permanent restricted status for a product.

For internally metered products, sellers need to call separate change types AddDimensions and UpdatePricingTerms to update pricing for the instance type.

To add new instance types, call the StartChangeSet API operation with the RestrictRegions change type, as shown in the following example.

Request Syntax

Only AddInstanceTypes change type is shown below. Although internally metered AMI sellers are required to call AddInstanceTypes and UpdatePricingTerms change types for their AMI.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
```

```

    "ChangeType": "AddInstanceTypes",
    "Entity":
    {
      "Identifier": "prod-example12345",
      "Type": "AmiProduct@1.0"
    },
    "DetailsDocument":
    {
      "InstanceTypes":
      [
        "m1.medium"
      ]
    }
  ]
}

```

Provide information for the fields to add the AddInstanceTypes change type:

- **Entity** (object) (required) – Your AMI-based product.
 - **Identifier** (string) (required) – Your product ID. For more information, see [Identifier](#).
 - **Type** (string) (required) – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`.
- **DetailsDocument** (object) (required) – The details required to execute the ChangeSet, in this case InstanceTypes.
 - **InstanceTypes** (array of strings) (required) – List of InstanceTypes to add to the product. These instances will be added to the existing InstanceTypes.

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```

{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}

```


The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to AddInstanceTypes actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP
InstanceTypes	Required	422
InstanceTypes	Must not be empty	422
InstanceTypes	Entries must be between 1 to 24 characters long. Must match <code>^[A-Za-z0-9_-.]+\$</code>	422
InstanceTypes	Entries must be unique	422
InstanceTypes	Must not be more than 1500 entries	422
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to AddInstanceTypes actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_INSTANCE_TYPES	The following instance types are not valid: [x]

Error code	Error message
INVALID_INSTANCE_VIRTUALIZATION	Virtualization of the product is [x]. It is not supported by the following instance types: [x]
INVALID_AMI_ARCHITECTURE	CPU architecture of the product is '%s'." + "It is not supported by the following instance types: [x]
INCOMPATIBLE_OPERATING_SYSTEM	The instance types are incompatible with the OS defined in the product. Provide instance types that are compatible with the OS defined in the product.
INVALID_PRODUCT_TYPE	Use an existing single AMI product.
INVALID_ENA_SETTING	The product does not support ENA. ENA support is required by the following instance types: [x]
INVALID_DIMENSIONS	No internally metered dimensions found for instance types: [x]
UPDATE_PRICING_REQUIRED	UpdatePricingTerms change type is required when internally metered dimensions are available on the product.

Restrict an instance type

You can use the Catalog API to limit or restrict the instance types available for your AMI-based product in AWS Marketplace.

Note

For more information about limiting or restrict the instance types available using the AWS Marketplace Management Portal, see [Restrict an instance](#) in the *AWS Marketplace Seller Guide*.

Existing subscribers won't be impacted by this change and they are able to use the restricted instance types. However, no new buyers will be able to use restricted instance types. To stop current instance types subscriptions (once instance types are restricted), you must contact the AWS Marketplace Seller Operations Team.

`RestrictInstanceTypes` restricts instance types to all the versions in the product document. In the `AddInstanceTypes` change type, you are updating all versions of the product. You won't be able to restrict the recommended instance types. The recommended instance type is at the version level, so it's possible that the seller won't be able to restrict multiple instance types.

For an internally metered product, you need to call separate change types when calling `RestrictDimensions`. This prevents new offers being created for the restricted instance types.

Note

The `RestrictInstanceTypes` change type is only available on `AmiProduct@1.0`.

To limit or restrict the instance types available for your AMI-based product, call the `StartChangeSet` API operation with the `RestrictInstanceTypes` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "RestrictInstanceTypes",
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument":
      {
        "InstanceTypes":
```

```
    [
      "m1.medium"
    ]
  }
}
]
```

Provide information for the fields to add the `RestrictInstanceTypes` change type.

- **Entity (object) (required)** – Your AMI-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `AmiProduct@1.0`.
- **DetailsDocument (object) (required)** – The details required to execute the `ChangeSet`, in this case `InstanceTypes`.
 - **InstanceTypes (array of strings) (required)** – List of `InstanceTypes` to restrict to the product. These instances are added to the current (or if there are no existing instance types, it will add) to restricted `InstanceTypes`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This included validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `RestrictInstanceTypes` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP
InstanceTypes	Required	422
InstanceTypes	Must not be empty	422
InstanceTypes	Entries must be between 1 to 24 characters long. Must match <code>^[A-Za-z0-9_-.]+\$</code>	422
InstanceTypes	Entries must be unique	422
InstanceTypes	Must not be more than 1500 entries	422
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to `RestrictInstanceTypes` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_INSTANCE_TYPES	The following instance types are not valid: [x]
INVALID_PRODUCT_TYPE	Use an existing single AMI product.
DUPLICATE_INSTANCE_TYPE	Provide instance types with no duplicates.
UNAVAILABLE_INSTANCE_TYPE	Provide an available instance type.
RECOMMENDED_INSTANCE_TYPE_RESTRICTED	The following instance types cannot be restricted. Recommended instance type must

Error code	Error message
	be changed to a different one before being restricted. Delivery Options Id [X] Instance Type[X]
DIMENSIONS_NOT_RESTRICTED	Restrict dimensions before restricting internally metered instance types: [x]
REGION_NO_INSTANCES	Your restricted instance types would cause product launch failure in region: X. Consider restricting fewer instances.

Work with EC2 Image Builder component products

As an AWS Marketplace seller, you can list AMI-based products delivered to AWS Marketplace buyers using EC2 Image Builder components. To create your component and publish an AWS Marketplace listing, proceed sequentially through the following sections.

Topics

- [Building and testing your Image Builder component](#)
- [Copying the component ARN](#)
- [Creating AWS Marketplace IAM policies](#)
- [Creating the AWS Marketplace IAM role](#)
- [Prepare your Image Builder component listing](#)
- [Publishing your Image Builder component product listing](#)

Building and testing your Image Builder component

Build and test your component on Image Builder. For instructions, refer to [Develop custom components for your Image Builder image](#) in the *Image Builder User Guide*. When creating your component using Image Builder, ensure that you do the following:

- The component and all of its underlying dependencies, such as an Amazon Simple Storage Service (Amazon S3) bucket, secrets, or parameters, must be created in the US East (N. Virginia) (us-east-1) AWS Region.

- Include supported architecture and any software dependencies in the component description.
- Test your component in your AWS account by creating an [image pipeline](#) and deploying the AMI created by the pipeline.
- If your component contains instructions to copy binaries, packages, or files from an S3 bucket, use the S3Download action module. In the S3Download module, for source, enter the static location of your file in the S3 bucket. The following example copies a binary from an S3 bucket as part of the component installation.

```
- name: DownloadMyFile
  action: S3Download
  inputs:
    - source: s3://amzn-s3-demo-source-bucket/path/to/package.zip
      destination: C:\myfolder\package.zip
```

- Components can ingest files of up to 2 GB when using the S3Download action.
- If your component uses [parameters](#), ensure that all parameters have default values. For example, if you have a parameter named `region`, ensure that you have a valid default value such as `us-east-1`. These default values are for AWS Marketplace processing and testing. Testing may fail if you do not include default values.
- If your component uses AWS Secrets Manager, Parameter Store, or a capability of AWS Systems Manager to store parameters, do the following:
 - To retrieve values as a step in your component, embed AWS Command Line Interface commands in your YAML configuration file.
 - Create a corresponding entry in Secrets Manager or Parameter Store in your AWS account. Use the default key and provide a valid value that will assist in building the component during the AWS Marketplace testing process. For example, say you have a parameter called `saas_token` with a default value of `token` that uses Parameter Store. In this case, create a key-value pair in Parameter Store. Use `token` as the key. For the value, enter a valid SaaS token for your application.

Note that values stored in your AWS Marketplace seller account will only be used for AWS Marketplace testing purposes. These values will not be shared with buyers.

- AWS Marketplace automatically generates Amazon Machine Images (AMIs) for your component across all compatible operating system versions you choose during the component creation process. When building your component, choose at least one compatible operating system

version. Validate your component's compatibility with all chosen operating system versions by using EC2 Image Builder pipelines to create and test AMIs.

Copying the component ARN

After creating and testing the component on Image Builder, copy and save the component ARN. You will use the ARN when you publish the product listing using the AWS Marketplace Catalog API.

To copy the Image Builder component ARN

1. Sign in to the AWS Management Console and open the Image Builder console at <https://console.aws.amazon.com/imagebuilder/>.
2. In the left navigation bar, under **Saved resources**, choose **Components**.
3. On the **Components** page, for **Filter owner**, choose **Owned by me**.
4. Choose the component name.
5. On the component detail page, in the **Summary** section, copy the ARN.

Creating AWS Marketplace IAM policies

Create the following IAM policies to grant AWS Marketplace access to your Image Builder component and related resources like Amazon S3 buckets and secrets. Use the example policies provided. You attach these policies to an [AWS Marketplace IAM role](#). For help creating policies, see [Creating policies using the JSON editor](#) in the *IAM User Guide*.

- Image Builder get-component policy, to allow AWS Marketplace to access your component on Image Builder. This policy is required. Name the policy mp_ib_ingest.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "imagebuilder:GetComponent",
      "Resource": "*"
    }
  ]
}
```


- Amazon S3 read-access policy, to allow AWS Marketplace to retrieve binaries from an S3 bucket. This policy is only required if your component uses the S3Download action module and stores associated binaries in an S3 bucket. Name the policy mp_ib_s3_read_only.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListObjectsInBucket",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::bucket_name"
      ]
    },
    {
      "Sid": "ReadObjectsInBucket",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectAttributes"
      ],
      "Resource": [
        "arn:aws:s3:::bucket_name/*"
      ]
    }
  ]
}
```

- Secrets Manager read-access policy, to allow AWS Marketplace to retrieve secrets stored in Secrets Manager. This policy is only required if your component uses Secrets Manager to store secrets. Name the policy mp_ib_sm_read_only. To restrict the policy to just your secret, replace the * in the Resource section with your secret.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
```

```

        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": [
            "*"
        ]
    }
]
}

```

- Parameter Store read-access policy, to allow AWS Marketplace to retrieve secrets stored in Parameter Store. This policy is only required if your component uses Parameter Store to store secrets. Name the policy `mp_ib_ssm_parameter_read_only`. To restrict the policy to just your secret, replace the `*` in the Resource section with your secret.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Action": [
        "ssm:getParameter"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}

```

Creating the AWS Marketplace IAM role

Use the following procedure to create an AWS Marketplace IAM role with policies to grant AWS Marketplace access to your component and its dependencies.

To create the AWS Marketplace IAM role

1. Sign in to the AWS Management Console and open the IAM console at <https://console.aws.amazon.com/iam/>.
2. In the left navigation bar, choose **Roles**.

3. Choose **Create role**.
4. Select **Custom trust policy**.
5. Enter the following statement:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Principal": {
        "Service": "assets.marketplace.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

6. Choose **Next**.
7. Add the Image Builder get-component policy you created previously. The get-component policy is required. Add the relevant policies for S3, Secrets Manager, and Parameter Store if your component uses these AWS services.
8. Choose **Next**.
9. Enter a role name, such as `MPEC2IBIngestion`.
10. Choose **Create role**.

Copy AWS Marketplace IAM role ARN

After creating the AWS Marketplace IAM role, copy and save the role ARN. You will use the ARN when publishing the listing using the AWS Marketplace Catalog API.

To copy the AWS Marketplace IAM role ARN

1. In the IAM console, in the left navigation bar, choose **Roles**.
2. Choose the AWS Marketplace IAM role you created previously, such as `MPEC2IBIngestion`.
3. On the role detail page, in the **Summary** section, copy the ARN.

Prepare your Image Builder component listing

Before publishing your AWS Marketplace listing, ensure that you have the following information ready:

- **Product metadata** – Metadata includes the product logo, product title, End User License agreement, supported instance types, and AWS Region.
- **Pricing information** – You can offer your product for free, at an hourly rate, or at an hourly rate with an initial free trial period. Bring your own license (BYOL) is not supported.
- **Component details** – Details include the component Amazon Resource Number (ARN), usage details, and AWS Identity and Access Management (IAM) role that AWS Marketplace will assume to process your component.

Publishing your Image Builder component product listing

This topic contains instructions to publish your EC2 Image Builder component listing on AWS Marketplace using the AWS Marketplace Catalog API.

Prerequisites

Ensure that you have the following before publishing your Image Builder component product listing:

- Registration as a seller in AWS Marketplace. For more information, refer to [Register as an AWS Marketplace seller](#).
- An IAM user with `AWSMarketplaceSellerFullAccess` permission.
- A publicly accessible Amazon Simple Storage Service (Amazon S3) bucket to host your company logo and EULA, if used in your component. You will enter the URL for the S3 bucket in your ChangeSet JSON file.
- AWS Command Line Interface (AWS CLI). For more information, refer to [What is the AWS Command Line Interface?](#) in the *AWS Command Line Interface User Guide*.

Creating an EC2 Image Builder component product

To create an EC2 Image Builder component product on AWS Marketplace using the Catalog API, refer to [Create a product](#).

Work with container-based products using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with container-based products.

For information about creating a container-based product using the Catalog API, see [Create a product](#).

The following topics describe how to use the Catalog API to perform actions on your container-based products:

Topics

- [Add a new version](#)
- [Update the visibility for an Amazon EKS add-on](#)
- [Create repositories and resources](#)
- [Update version information](#)
- [Restrict a version](#)

Add a new version

If you already have a container-based product in AWS Marketplace, you can use the AWS Marketplace Catalog API to add a new version. This requires that you have already created repositories in AWS Marketplace for each container image or artifact that is part of your product, and that you can copy them from your local Docker and Helm files.

Note

For details about creating a container-based product using the AWS Marketplace Management Portal, see [Getting started with container products](#) in the *AWS Marketplace Seller Guide*.

For details about adding a new version, including creating repositories and building Docker and Helm files into those repositories, by using the AWS Marketplace Management Portal, see [Add a new version of your product](#) in the *AWS Marketplace Seller Guide*.

If you have not already created new repositories, you can create them using the Catalog API, see [Create repositories and resources](#).

To add a new version, call the `StartChangeSet` API operation with the `AddDeliveryOptions` change type, as shown in the following example.

Note

A version of a container-based product is made up of one or more delivery options. For example, you might have two delivery options, one that works with a noSQL database, and another that works with MySQL, so that your users can choose how they want to work with your product. You create the version of your product and add multiple delivery options in a single request with `AddDeliveryOptions`.

Container Image Delivery Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity":
      {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument":
      {
        "Version":
        {
          "VersionTitle": "1.1",
          "ReleaseNotes": "Minor bug fix"
        },
        "DeliveryOptions":
        [
          {
            "DeliveryOptionTitle": "EKS Container image only delivery option",
            "Details":
            {
              "EcrDeliveryOptionDetails":
```



```

    "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
    "Type": "ContainerProduct@1.0"
  },
  "DetailsDocument":
  {
    "Version":
    {
      "VersionTitle": "1.1",
      "ReleaseNotes": "Minor bug fix"
    },
    "DeliveryOptions":
    [
      {
        "DeliveryOptionTitle": "HelmChartDeliveryOption",
        "Details":
        {
          "HelmDeliveryOptionDetails":
          {
            "CompatibleServices":
            [
              "EKS",
              "EKS-Anywhere"
            ],
            "ContainerImages":
            [
              "111122223333.dkr.ecr.us-east-1.amazonaws.com/sellername/
reponame1:1.1"
            ],
            "HelmChartUri": "111122223333.dkr.ecr.us-east-1.amazonaws.com/
sellername/reponame1:helmchart1.1",
            "Description": "Helm chart description",
            "UsageInstructions": "Usage instructions",
            "QuickLaunchEnabled": true,
            "MarketplaceServiceAccountName": "Service account name",
            "ReleaseName": "Optional release name",
            "Namespace": "Optional Kubernetes namespace",
            "OverrideParameters":
            [
              {
                "Key": "HelmKeyName1",
                "DefaultValue": "${AWSMP_LICENSE_SECRET}",
                "Metadata":
                {
                  "Label": "AWS CloudFormation template field label",

```



```

    {
      "DeliveryOptionTitle": "AWS Marketplace Test AddOn from CAPI 1",
      "Visibility": "Limited",
      "Details": {
        "EksAddOnDeliveryOptionDetails": {
          "ContainerImages": [
            "111122223333.dkr.ecr.us-east-1.amazonaws.com/test-seller/canary-
test-repo-product-6:mongo"
          ],
          "HelmChartUri": "111122223333.dkr.ecr.us-east-1.amazonaws.com/rocket/
rocket-product-helm:1.0",
          "Description": "Description for delivery option provided by ISV",
          "UsageInstructions": "Usage instructions with launch instructions",
          "AddOnName": "aws-mp-test",
          "AddOnVersion": "1.2.1",
          "AddOnType": "networking",
          "CompatibleKubernetesVersions": [
            "1.25",
            "1.26"
          ],
          "SupportedArchitectures": [
            "amd64",
            "arm64"
          ],
          "Namespace": "my-test-namespace",
          "EnvironmentOverrideParameters": [
            {
              "Key": "cluster-name",
              "Value": "${AWS_EKS_CLUSTER_NAME}"
            },
            {
              "Key": "region-name",
              "Value": "${AWS_REGION}"
            }
          ]
        }
      }
    },
    "ChangeName": "PublishAddonNew"
  ]
}

```

```
}
```

Provide information for the fields to add the `AddDeliveryOptions` change type:


- **Entity (object) (required)** – Your container-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `ContainerProduct@1.0`.
- **DetailsDocument (object) (required)** – Details of the request. It includes all the information about the version that you are adding. This field is a string field.
- **Version (object)** – Details about the version that you are adding to your product.
 - **VersionTitle (string)** – The title of the version that you are creating. Typically this is a description of the version, like **Version 1.1** or simply **1.1**. Your buyers will be able to choose the version to deploy from a list of version titles.
 - **ReleaseNotes (string)** – The detailed notes about this version. Must be less than 30,000 characters.
- **DeliveryOptions (array of objects)** – An array of delivery options, where each is a method of delivery for your product version. For example, if you have one delivery option for Amazon Elastic Container Service (Amazon ECS) and another for Amazon Elastic Kubernetes Service (Amazon EKS), you will need to have two delivery options.
 - **DeliveryOptionTitle (string)** – A short description that helps your buyer to choose between your delivery options.
 - **Details (object)** – The resources used for this delivery option. This is a details field within the details field. You do not need to doubly escape characters in this field.
 - **AddOnName** – A unique add-on name that buyers will see in the Amazon EKS catalog. This name will add a prefix later using `SellerAlias`. For example, where `isv-alias_` is the ISV provided add-on name.
 - **AddOnType** – The type of add-on chosen from a list of supported values from Amazon EKS: `Gitops | monitoring | logging | cert-management | policy-management | cost-management | autoscaling | storage | kubernetes-management | service-mesh | etcd-backup | ingress-service-type | load-balancer | local-registry | networking | Security | backup | ingress-controller | observability`
 - **AddOnVersion** – A semantic version so that buyer can choose a specific version of `AddOn` they need to install or upgrade.

- `CompatibleKubernetesVersions` – The Amazon EKS Kubernetes versions that this software is compatible with.
- `CompatibleServices` (array of strings) – An array of services that the release is compatible with. Valid options are ECS and EKS.
- `ContainerImages` (array of strings) – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the tag for the image used by this version. The list must include all needed images, even images that have not changed from previous versions. See the next section for information about creating repositories using the Catalog API.
- `Description` (string) – A longer description of the delivery option to give details to your buyer. You can also include a link to more instructions provided elsewhere.
- `EcrDeliveryOptionDetails` – `DeploymentResources` (array of objects) – An array of other resources needed for the version, such as Helm charts. Each resource includes a `Name` to describe it, and a `URL` that points at the resource.
- `EnvironmentOverrideParameters` – List of system parameters to be used by the add-on. Some of the ISV provided AddOn (HelmChart) might require configurations with information derived from the Amazon EKS execution environment state (/system information). For example, `EksClusterRegion`, `EKSClusterName`, and others. You can avoid additional actions from Buyer by dynamically substituting these values at Amazon EKS AddOn launch. Amazon EKS System already supports automatic substitutions of system param for add-ons. AWS Marketplace ISV experience can be extended to collect this params which would require substitution.

The generic system information to be substituted can be indicated by providing a AWS Marketplace specified constant following convention similar to Helm substitution. The supported values are `${AWS_REGION}` and `${AWS_EKS_CLUSTER_NAME}`.

```
"EnvironmentOverrideParameters" : [ {
  "Key" : "my-field.region"
  "Value" : "${AWS_REGION}"
},
{
  "Key" : "my-second-field"
  "Value" : "${AWS_EKS_CLUSTER_NAME}"
},
```

- `HelmDeliveryOptionDetails - HelmChartUri` (string) – The URL to the Helm chart hosted in Amazon ECR that the buyer will install to launch the software.
- `HelmDeliveryOptionDetails - QuickLaunchEnabled` (boolean) – A boolean to determine if buyers can use QuickLaunch to launch the software. For more information about QuickLaunch, see [QuickLaunch in AWS Marketplace](#).
- `HelmDeliveryOptionDetails - MarketplaceServiceAccountName` (string) – *Optional* – The name of the Kubernetes service account. The service account will be used to connect to AWS Identity and Access Management (IAM) for permissions to call AWS services.
- `HelmDeliveryOptionDetails - ReleaseName` (string) – *Optional* – The name for the Helm release provided to the `helm install` command that buyers use to launch the software. If not included, Helm will provide an automatically generated release name for you.
- `HelmDeliveryOptionDetails - Namespace` (string) – *Optional* – The Kubernetes namespace where the Helm chart will be installed.
- `HelmDeliveryOptionDetails - OverrideParameters` (array of objects) – Parameters that will be used in the Helm commands that launch the application. Buyers can override the default values.

 **Note**

For Amazon EKS Anywhere products, provide at least 1 override parameter for the license secret. Provide `DefaultValue` of `"${AWSMP_LICENSE_SECRET}"`. For paid products, provide at least 1 override parameter for service account configuration. Provide `DefaultValue` of `"${AWSMP_SERVICE_ACCOUNT}"`.

- `Key` (string) – The key for the parameter in dot notation (`override.example.key`).
- `DefaultValue` (string) – The default value for this override parameter.
- `Metadata` (array of objects) – *Required if QuickLaunchEnabled is set to **true*** – An array of objects that include details about the override parameter, including AWS CloudFormation template information.
 - `Label` (string) – The name of the field in the AWS CloudFormation stack creation form that buyers use during QuickLaunch.

- **Description** (string) – The description of the field in the AWS CloudFormation stack creation form that buyers use during QuickLaunch.
- **Obfuscate** (boolean) – A boolean to determine if sensitive information such as secrets and passwords are masked in AWS CloudFormation consoles, commands, and APIs.
- **Namespace** – The ISV provided namespace for add-on installation.
- **SupportedArchitectures** – The list of supported architectures, like amd64 and arm64.
- **UsageInstructions** (string) – Provide instructions about the usage for this delivery option. Can be up to 4,000 characters.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed, including scanning the container images and other information to ensure that it meets the [AWS Marketplace guidelines for container products](#). This process can take a few minutes to hours, depending on the number and size of your containers.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

Asynchronous Errors

The following errors are specific to `AddDeliveryOptions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is

processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT_STATUS	Use an existing limited or public product.
INCOMPATIBLE_SERVICES	The service list contains incompatible services. [incompatible_services] Provide a valid list of compatible services.
NO_SERVICE_SPECIFIED	Provide at least 1 compatible service.
DUPLICATE_COMPATIBLE_AWS_SERVICES	The service list contains duplicate entries. Remove them. Each entry must be unique.
INVALID_VERSION_TITLE	Remove spaces before the trademark symbol.
INVALID_VERSION_TITLE	Remove the following unsupported characters: [x, y, z]
INVALID_VERSION_TITLE	Remove spaces from the beginning of the version title.
INVALID_VERSION_TITLE	Provide version title with fewer than [x] characters.
DUPLICATE_VERSION_TITLE	The version title [duplicate_version_title] is a duplicate. Remove or change the title.
INVALID_RELEASE_NOTES	Remove spaces before the trademark symbol.
INVALID_RELEASE_NOTES	Remove unsupported characters: [x, y, z]
INVALID_RELEASE_NOTES	Remove spaces from the beginning of release notes.
INVALID_RELEASE_NOTES	Provide release notes with fewer than (x) characters.

Error code	Error message
INVALID_USAGE_INSTRUCTIONS	Remove spaces before the trademark symbol.
INVALID_USAGE_INSTRUCTIONS	Remove unsupported characters: [x, y, z]
INVALID_USAGE_INSTRUCTIONS	Provide usage instructions with fewer than (x) characters.
INVALID_USAGE_INSTRUCTIONS	Provide usage instructions.
MISSING_CONTAINER_IMAGES	Provide at least 1 container image.
NO_LICENSE_SECRET_KEYS	For Amazon EKS Anywhere products, provide 1 override parameter for license secret. Needs <code>DefaultValue</code> of " <code>\${AWSMP_LICENSE_SECRET}</code> " , see example in section.
TOO_MANY_CONTAINER_IMAGES	Provide fewer than 50 container images.
DUPLICATE_CONTAINER_IMAGES	The container image list contains duplicate images: [duplicate_images]. Provide a list with unique images.
INVALID_CONTAINER_IMAGES	Provide a valid URI for the container image.
INVALID_CONTAINER_IMAGE_URI	The image [invalid_image_uri] doesn't have access to this product. Upload the image to its corresponding product repository. For information about uploading, see Getting started with container products .
INVALID_CONTAINER_IMAGE_TAG	Avoid using 'latest' tag.
DUPLICATE_DELIVERY_OPTION_TITLES	Duplicate delivery option titles: [duplicate_titles]. Remove the duplicates
INVALID_DELIVERY_OPTION_TITLES	The delivery option titles [existing_titles] already exist. Provide a different title.

Error code	Error message
INVALID_FULFILLMENT_OPTION_TITLE	Provide delivery option title with fewer than (x) characters.
NO_SERVICE_ACCOUNT_CONFIGURATION	For paid products, provide 1 override parameter for service account configuration. Needs DefaultValue of "\${AWSMP_SERVICE_ACCOUNT}" , see example in section.
INVALID_DETAILS	Provided Details is not valid.
EMPTY_RESOURCE_NAME	Provide resource name.
EMPTY_RESOURCE_URL	Provide resource URL.
INVALID_RESOURCE_NAME	Provide resource name with fewer than 256 characters.
INVALID_RESOURCE_URL	Provide resource URL with fewer than 256 characters.
INVALID_SHORT_DESCRIPTION	Provide a short description with fewer than 1,000 characters.
INVALID_SHORT_DESCRIPTION	Provide short description.
SCAN_ERROR	Fix security vulnerability ""[y]"" on Image ""[x]"".
IMAGE_NOT_FOUND	The public image URI [invalid_image_uri] is invalid. Provide a valid URI.
INVALID_ARN	Provide a valid ARN for image access.
IMAGE_INACCESSIBLE	Provide a valid ARN for image access.

Error code	Error message
DUPLICATE_ADDON_NAME	The AddOn name you provided is already in use by a different product. Provide a different name.
DUPLICATE_ADDON_VERSION	The add-on version title [duplicate_version_title] is already in use. Provide a different title.
INVALID_ADDON_TYPE	The add-on types [invalid_types] are invalid. Provide a type from the supported list: [eks_addon_do_supported_types].
INVALID_KUBERNETES_VERSION	The Kubernetes versions [invalid_versions] are invalid. Provide versions from the supported list: [eks_addon_do_supported_kubernetes_versions].
DUPLICATE_KUBERNETES_VERSIONS	Duplicate Kubernetes versions: [duplicate_versions]. Provide a list with unique versions.
INVALID_ARCHITECTURE	The architectures [invalid_architectures] are invalid. Provide architectures from the Amazon EKS supported architectures : [eks_addon_do_supported_architectures].
DUPLICATE_SUPPORTED_ARCHITECTURES	Duplicate architectures: [duplicate_architectures]. Provide a list of unique, supported architectures.
INVALID_VISIBILITY_STATE	The states [invalid_states] are invalid for the {EKS_DO} delivery option. Provide a valid visibility state from the following allowed values: Limited.

Error code	Error message
INVALID_ENVIRONMENT_OVERRIDE_PARAMETER_VALUE	The override parameter values [invalid_values] are invalid. Provide a valid value from the following list: [eks_addon_do_environment_override_parameter_values].
DUPLICATE_ENVIRONMENT_OVERRIDE_PARAMETER_KEY	The environment override parameters contain duplicate keys: [duplicate_keys]. Remove them.
TOO_MANY_EKS_ADDON_DELIVERY_OPTIONS	Provide only one Amazon EKS add-on delivery option for the version.
INCOMPATIBLE_ADDON_NAME	The add-on name [provided_name] does not match the existing name. Reuse the existing name from the public version, or previous versions, of this add-on. You can only use one add-on name for each product.
INCOMPATIBLE_ADDON_TYPE	The add-on type [provided_type] does not match the existing type. Reuse the existing type from the public version, or previous versions, of this add-on. You can only use one add-on type for each product.
INCOMPATIBLE_ADDON_NAMESPACE	The add-on namespace [provided_namespace] does not match the existing namespace . Reuse the existing namespace from the public version, or previous versions, of this add-on. You can only use one add-on namespace for each product.
INVALID_HELM_CHART_URI	The Helm chart URI [invalid_uri] is invalid. Provide a URI in the SemVer 2 format.

Error code	Error message
INCOMPATIBLE_HELM_OBJECTS(INVALID_HELM_OBJECTS)	Provide a Helm chart without using the following unsupported Helm Objects: <unsupported-objects>.
INVALID_DEPENDENT_HELM_CHARTS	Provide a Helm chart that contains the following dependent charts directly in the parent chart directory and not externally sourced: <invalid-subcharts>.
INVALID_HELM_SENSITIVE_CONFIG	Provide an advanced configuration schema without sensitive information or secrets. Keywords: <sensitive-parameters-identified>
INVALID_HELM_UNDECLARED_IMAGES	Provide the following Helm chart images within the delivery option of the request: <list-of-images>.
INVALID_HELM_CHART_IMAGES	Provide a Helm chart containing images within repositories created via the AddRepositories change type. External images: <images-identified>.
INVALID_HELM_LINT	Provide a Helm chart that successfully passes Helm lint.
INVALID_HELM_TEMPLATE	Provide a Helm chart that successfully passes Helm template.
INVALID_HELM_CHART	Provide a Helm chart that adheres to AWS Marketplace guidance identified in Helm Charts bulleted list in the <i>AWS Marketplace Seller Guide</i> .
INVALID_ADDON_NAME	Provide an AddOn name that follows the following regex pattern: xx

Error code	Error message
INVALID_ADDON_NAMESPACE	The namespace values [invalid_namespace s] are invalid. The namespace must follow the {EKS_ADD_ON_NAMESPACE_REGEX} regular expression. For example, namespace, namespace-test.
INVALID_ADDON_NAME_PATTERN	Provide an add-on name that starts with a letter or digit, and then a combination of letters, digits, and hyphens. For example, test-addon, eksaddon
INVALID_ADDON_VERSION_PATTERN	Provide an add-on version using the following pattern: "<major>.<minor>.<patch>" (for example, 1.2.3, 0.1.2, 0.1.1)
EMPTY_DELIVERY_OPTION_IDS	Provide a list of delivery option IDs.
INVALID_DELIVERY_OPTIONS_INPUT	The list contains one or more invalid delivery options. Provide a valid list, and ensure that each option has a single delivery method.
OVERRIDE_PARAMETER_KEYS_CONTAINS_SPECIAL_CHARS	The override parameter keys [invalid_keys] contain invalid characters. Your keys must use only letters, numbers, double quotes (" ") and plus signs (+).
INVALID_CONTAINER_IMAGE_REPOSITORY	The repositories [invalid_repositories] are invalid. Provide repositories created through the AddRepositories change type.
INVALID_CONTAINER_IMAGE_TAG_FORMAT	The container image tag [invalid_image_tag] is invalid. Provide a tag conforming to the {CONTAINER_IMAGE_TAG_REGEX} regular expression.
DUPLICATE_OVERRIDE_PARAMETER_KEYS	The override parameters contain duplicate keys [duplicate_keys]. Remove the duplicates.

Error code	Error message
UNSUPPORTED_CONTAINER_IMAGE_URI	The container image [unsupported_image] is unsupported. Provide an image that follows the https://docs.docker.com/registry/spec/manifest-v2-1/#example-manifest schema.
DUPLICATE_REPOSITORY_NAMES	Duplicate repository names: [duplicate_repo_names]. Provide unique names.
INVALID_NAMESPACE	The namespace values [invalid_namespaces] are invalid. Provide values that conform to the {HELM_RELEASE_PARAM_REGEX} regular expression.
INVALID_RELEASE_NAME	The releaseName values [invalid_release_names] are invalid. Provide values that conform to the {HELM_RELEASE_PARAM_REGEX} regular expression.
OVERRIDE_PARAMETER_KEYS_CONTAINS_RESERVED_PARAMETER_KEYS	The override parameter key for delivery option titles [invalid_keys] is reserved. Reserved keys: [reserved_param_keys]. Provide a different key.

Update the visibility for an Amazon EKS add-on

You can use the Catalog API to update visibility for an Amazon EKS add-on delivery option of your product version in AWS Marketplace. Container and Helm delivery options for your container product are automatically created with 'Public' visibility status.

Note

The ability to update visibility of your product version is supported only for the Amazon EKS add-on delivery option from the listed versions. If your product isn't 'Public' already, submit a request to publish the product with 'Public' visibility status by using the AWS Marketplace Management Portal.

By default, when you create a version with the Amazon EKS add-on delivery option, it's published in 'Limited' status. A 'Limited' status means the product isn't publicly available across all the Regions for your buyers to use and deploy in an Amazon EKS cluster. You can update the visibility of the delivery option from 'Limited' to 'Public' by calling the `StartChangeSet` API operation with the `UpdateDeliveryOptionsVisibility` change type. Specify the `DeliveryOptions Id` from your product version that corresponds to the Amazon EKS add-on delivery option.

Request Syntax

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateDeliveryOptionsVisibility",
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument":
      {
        "DeliveryOptions":
        [
          {
            "Id": "do-1234567891234567891234",
            "TargetVisibility": "Public"
          }
        ]
      }
    }
  ]
}
```

To add the `UpdateDeliveryOptionsVisibility` change type, provide information for the following fields :

- **Entity (object) (required)** – Your container-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).

- **Type** (string) (required) – The Type is based on the delivery method (product type) that your product uses: `ContainerProduct@1.0`.
- **DetailsDocument** (object) (required) – Details of the request, including the information about the repositories that you want to create. The following fields are all required.
- **DeliveryOptions** (list of objects) – List of `DeliveryOption` objects, including the details of each:
 - **Id** (string) – Unique identifier for the `DeliveryOption`. (To get the unique identifier for the `DeliveryOption`, call the `DescribeEntity` action on the product that you're updating.
 - **TargetVisibility** – The intended new visibility of the product.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed, including scanning the container images and other information to ensure that it meets the [AWS Marketplace guidelines for container products](#). This process can take a few minutes to hours, depending on the number and size of your containers.

You can check the status of the request through the AWS Marketplace Management Portal, or through the AWS Marketplace Catalog API by using the [DescribeChangeSet](#) API operation.

For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

Asynchronous Errors

The following table shows errors that are specific to `AddDeliveryOptions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a

change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
EMPTY_DELIVERY_OPTION_IDS	Provide a list of delivery option IDs.
INVALID_VISIBILITY_STATE	The <code>TargetVisibility</code> option you provided is not supported. Please try again with an allowed option. The allowed option(s) are: <code>Public</code>
INVALID_DELIVERY_OPTION_IDS	You provided invalid delivery option details. Provide delivery option IDs that can be found in the product. IDs not found: [x]
DUPLICATE_DELIVERY_OPTION_IDS	Duplicate delivery option IDs: [duplicate_ids]. Provide unique delivery option IDs.
AUDIT_ERROR	You haven't completed independent software vendor (ISV) testing for all compatible Amazon EKS cluster versions for your Amazon EKS add-on version(s). You must complete testing before we can release the delivery option(s).
INVALID_DELIVERY_OPTION_TYPE	The delivery option type you provided is not valid. Ensure that your delivery option is of type: <code>EksAddOn</code> and try again.
INCOMPATIBLE_HELM_OBJECTS	Provide a Helm chart without unsupported Helm Objects: Unsupported Helm objects are as follows: all Release objects (except <code>.Name</code> and <code>.Namespace</code>), Helm hooks, and lookup functions.

Error code	Error message
INCOMPATIBLE_ADDON_NAME	The add-on name [provided_name] does not match the public version name. Update the public name before releasing.
NCOMPATIBLE_ADDON_TYPE	The add-on types don't match. Reuse the existing add-on type from the public add-on version or previous add-on versions of this product. Only one add-on is supported for each product.
INCOMPATIBLE_ADDON_NAMESPACE	The provided add-on namespace [provided_namespace] does not match the public version namespace. Update the add-on namespace before releasing.

Create repositories and resources

To create a new version of a container-based product, you must have the resources for the version available in AWS Marketplace repositories. You create the repositories and then push (upload) the Docker (and Helm) resources into the repositories. To learn how to create the repositories through the AWS Marketplace Management Portal, see [Add a new version of your product](#) in the *AWS Marketplace Seller Guide*.

To create new repositories, call `StartChangeSet` with the `AddRepositories` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "AddRepositories",
```

```
"Entity":
{
  "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
  "Type": "ContainerProduct@1.0"
},
"DetailsDocument":
{
  "Repositories":
  [
    {
      "RepositoryName": "new-repo-1",
      "RepositoryType": "ECR"
    },
    {
      "RepositoryName": "new-repo-2",
      "RepositoryType": "ECR"
    }
  ]
}
]
```

Provide information for the fields to add the `AddRepositories` change type:

For more information about creating repositories, see [Adding a new version](#) in the *AWS Marketplace Seller Guide*.

- **Entity** (object) (required) – Your container-based product.
 - **Identifier** (string) (required) – Your product ID. For more information, see [Identifier](#).
 - **Type** (string) (required) – The Type is based on the delivery method (product type) that your product will be using: `ContainerProduct@1.0`.
- **DetailsDocument** (object) (required) – Details of the request. It includes the information about the repositories that you want to create. The included fields are all required.
 - **Repositories** (array of structures) – A list of repository objects. Each repository object includes a name and type.
 - **RepositoryName** (string) – The name of the repository to create.
 - **RepositoryType** (string) – The type of the repository to create. The only allowed value is `ECR`.

Note

You can have up to 50 repositories per product, although you can add multiple resources, and versions of resources, to a single repository by giving them different tags when you push them.

After you create one or more repositories, you add your resources to the repositories. For information about how to push resources to repositories, see [Pushing an image](#) in the *Amazon Elastic Container Registry User Guide*. For information about the specific push commands needed for one of your repositories, see [Adding a new version](#) in the *AWS Marketplace Seller Guide*.

Asynchronous Errors

The following errors are specific to `AddRepositories` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_ECR_REPOSITORY_NAME	Provide repository name in the format: 'nginx-web-app'
DUPLICATE_ECR_REPOSITORY_NAME	The repository [duplicate_repo_names] already exists. Choose a different name.
MISSING_REPOSITORY_INFORMATION	Provide at least 1 repository name.
INVALID_ECR_REPOSITORY_NAME	Maximum character length 256 reached. Character length count is inclusive of the seller namespace.

Update version information

You can use the Catalog API to update the details of an existing version of your container-based product in AWS Marketplace.

Note

When a product is publicly available, you cannot update the version title, container images, delivery option title, or deployment resources for the version. If you need to update these aspects of a product, create a new version instead.

To update an existing version of your container-based product, call the `StartChangeSet` API operation with the `UpdateDeliveryOptions` change type, as shown in the following example. This updates the detail information for the delivery options that you specify, as well as the associated version. You must include at least one delivery option.

Container Image Delivery Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "Version": {
          "ReleaseNotes": "New release notes",
          "VersionTitle": "Version 1.2"
        },
        "DeliveryOptions": [
          {
            "Id": "example4-2222-cccc-2222-cccccccccccc",
            "Details": {
              "EcrDeliveryOptionDetails": {
                "DeliveryOptionTitle": "New Delivery Option Title",
                "Description": "New description",
                "UsageInstructions": "New usage instructions",
                "CompatibleServices": [
                  "EKS"
                ]
              }
            }
          }
        ]
      }
    }
  ]
}
```



```

    "Description": "New description",
    "UsageInstructions": "New usage instructions",
    "MarketplaceServiceAccountName": "new-service-account-name",
    "ReleaseName": "new-release-name",
    "Namespace": "new-cluster-namespace",
    "QuickLaunchEnabled": true,
    "OverrideParameters": [
      {
        "Key": "new.parameter.key",
        "DefaultValue": "New parameter default value",
        "Metadata": {
          "Label": "New metadata label",
          "Description": "New metadata description",
          "Obfuscate": false
        }
      }
    ]
  }
}

```

Amazon EKS Add-On Delivery Request Syntax

```

POST /StartChangeSet HTTP/1.1
Content-type: application/json

```

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "Version": {
          "ReleaseNotes": "New release notes",

```

```

        "VersionTitle": "Version 1.2"
    },
    "DeliveryOptions": [
        {
            "Id": "example4-2222-cccc-2222-cccccccccccc",
            "Details": {
                "EksAddOnDeliveryOptionDetails": {
                    "ContainerImages": [
                        "709825985650.dkr.ecr.us-east-1.amazonaws.com/test-
seller/canary-test-repo-product-6:mongo"
                    ],
                    "Description": "Description for delivery option provided by
ISV",
                    "UsageInstructions": "Usage instructions with launch
instructions",
                    "HelmChartUri": "709825985650.dkr.ecr.us-
east-1.amazonaws.com/rocket/rocket-product-helm:1.0",
                    "AddOnName": "aws-mp-test",
                    "AddOnVersion": "1.2.1",
                    "AddOnType": "networking",
                    "CompatibleKubernetesVersions": [
                        "1.19",
                        "1.20"
                    ],
                    "SupportedArchitectures": [
                        "amd64",
                        "arm64"
                    ],
                    "Namespace": "my-test-namespace",
                    "EnvironmentOverrideParameters": [
                        {
                            "Key": "my-field",
                            "Value": "${AWS_EKS_CLUSTER_NAME}"
                        }
                    ]
                }
            }
        }
    ]
}

```


Provide information for the fields to add the `UpdateDeliveryOptions` change type:

For more information about these fields, see [Adding a new version](#) in the *AWS Marketplace Seller Guide*.

- `Entity` (object) (required) – Your container-based product.
 - `Identifier` (string) (required) – Your product ID. For more information, see [Identifier](#).
 - `Type` (string) (required) – The `Type` is based on the delivery method (product type) that your product will be using: `ContainerProduct@1.0`.
- `DetailsDocument` (object) (required) – Details of the request. It includes any information about the version of your container-based product that you would like to update. The included fields are all optional, but you must include at least one field to update.
- `Version` (object) – Details about the software version.
 - `VersionTitle` (string) – The title of the version that you are creating. Typically this is a description of the version, such as **Version 1.1** or simply **1.1**. Your buyers will be able to choose the version to deploy from a list of all version titles.

This property can't be updated if the product is already published publicly.

- `ReleaseNotes` (string) – Notes for buyers to tell them about changes from one version to the next.
- `DeliveryOptions` (list of objects) – List of `DeliveryOption` objects, including the details of each:
 - `Id` (string) – Unique identifier for the `DeliveryOption` (you can get the unique identifier for the `DeliveryOption` by calling the `DescribeEntity` action on the product you are updating).
 - `Details` (object) – Holds the details of a delivery option. Note that this nested details object does *not* need to be double-escaped.
 - `EcrDeliveryOptionDetails` (object) – The details of the container image delivery option.
 - `DeliveryOptionTitle` (string) – A short description that allows your buyer to choose between your delivery options.

This property can't be updated if the product is already published publicly.

- `ContainerImages` (array of strings) – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the

tag for the image used by this version. If this field is included, the list must include all needed images, even images that are not changing.

This property can't be updated if the product is already published publicly.

- `DeploymentResources` (array of objects) – An array of other deployment resources needed for the version, such as links to Helm charts or other documentation. Each resource includes a name to describe it and a URL that points at the resource. On the launch page for your version, this displays as a list of links.

This property can't be updated if the product is already published publicly.

- `Name` (string) – The text of the hyperlink that is shown to the buyer.
- `Url` (string) – The URL of the hyperlink shown to the buyer.
- `CompatibleServices` (array of strings) – A list of services that the release is compatible with. Valid options are ECS and EKS.
- `Description` (string) – A longer description of the delivery option to give details to your buyer. You can also include a link to more instructions hosted elsewhere.
- `UsageInstructions` (string) – Provide instructions on how to deploy and use your product. You can also add a link to usage instructions hosted elsewhere. Can be up to 4,000 characters.
- `Id` (string) – Unique identifier for the `DeliveryOption` (you can get the unique identifier for the `DeliveryOption` by calling the `DescribeEntity` action on the product you are updating).
- `Details` (object) – Holds the details of a delivery option. Note that this nested details object does *not* need to be double-escaped.
- `HelmDeliveryOptionDetails` (object) – The details of the Helm chart delivery option.
 - `DeliveryOptionTitle` (string) – A short description that allows your buyer to choose between your delivery options.

This property can't be updated if the product is already published publicly.

- `ContainerImages` (array of strings) – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the tag for the image used by this version. The list must include all needed images, even images that have not changed from previous versions. See the next section for information about creating repositories using the Catalog API.

- `HelmChartUri` (string) – The URL to the Helm chart hosted in Amazon ECR that the buyer will install to launch the software.
- `CompatibleServices` (array of strings) – An array of services that the release is compatible with. Valid options are ECS and EKS.
- `Description` (string) – A longer description of the delivery option to give details to your buyer. You can also include a link to more instructions provided elsewhere.
- `UsageInstructions` (string) – Provide instructions about the usage for this delivery option. Can be up to 4,000 characters.
- `MarketplaceServiceAccountName` (string) – The name of the Kubernetes service account. The service account will be used to connect to AWS Identity and Access Management for permissions to call AWS services.
- `ReleaseName` (string) – The name for the Helm release provided to the `helm install` command that buyers use to launch the software.
- `Namespace` (string) – The Kubernetes namespace where the Helm chart will be installed.
- `QuickLaunchEnabled` (boolean) – A boolean to determine if buyers can use QuickLaunch to launch the software. For more information about QuickLaunch, see [QuickLaunch in AWS Marketplace](#).
- `OverrideParameters` (array of objects) – Parameters that will be used in the Helm commands that launch the application. Buyers can override the default values.
 - `Key` (string)– The key for the parameter in dot notation (`override.example.key`).
 - `DefaultValue` (string) – The default value for this override parameter.
 - `Metadata` (array of objects) – *Only required if QuickLaunchEnabled is set to **true*** – An array of objects that include details about the override parameter, including AWS CloudFormation template information.
 - `Label` (string) – The name of the field in the AWS CloudFormation stack creation form that buyers use during QuickLaunch.
 - `Description` (string) – The description of the field in the AWS CloudFormation stack creation form that buyers use during QuickLaunch.
 - `Obfuscate` (boolean) – A boolean to determine if sensitive information such as secrets and passwords are masked in AWS CloudFormation consoles, commands, and APIs.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed, including scanning the container images and other information to ensure that it meets the [AWS Marketplace guidelines for container products](#). This process can take a few minutes to hours, depending on the number and size of your containers.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

Asynchronous Errors

The following errors are specific to `UpdateDeliveryOptions` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT_STATUS	Use an existing limited or public product.
INCOMPATIBLE_SERVICES	The service list contains incompatible services. [incompatible_services] Provide a valid list of compatible services.
NO_SERVICE_SPECIFIED	Provide at least 1 compatible service.

Error code	Error message
DUPLICATE_COMPATIBLE_AWS_SERVICES	The service list contains duplicate entries. Remove them. Each entry must be unique.
INVALID_VERSION_TITLE	Remove spaces before the trademark symbol.
INVALID_VERSION_TITLE	Remove the following unsupported characters: [x, y, z]
INVALID_VERSION_TITLE	Remove spaces from the beginning of the version title.
INVALID_VERSION_TITLE	Provide version title with fewer than [x] characters.
DUPLICATE_VERSION_TITLE	The version title [duplicate_version_title] is a duplicate. Remove or change the title.
INVALID_RELEASE_NOTES	Remove spaces before the trademark symbol.
INVALID_RELEASE_NOTES	Remove unsupported characters: [x, y, z]
INVALID_RELEASE_NOTES	Remove spaces from the beginning of release notes.
INVALID_RELEASE_NOTES	Provide release notes with fewer than (x) characters.
INVALID_USAGE_INSTRUCTIONS	Remove spaces before the trademark symbol.
INVALID_USAGE_INSTRUCTIONS	Remove unsupported characters: [x, y, z]
INVALID_USAGE_INSTRUCTIONS	Provide usage instructions with fewer than (x) characters.
INVALID_USAGE_INSTRUCTIONS	Provide usage instructions.
MISSING_CONTAINER_IMAGES	Provide at least 1 container image.
TOO_MANY_CONTAINER_IMAGES	Provide fewer than 50 container images.

Error code	Error message
DUPLICATE_CONTAINER_IMAGES	The container image list contains duplicate images: [duplicate_images]. Provide a list with unique images.
INVALID_CONTAINER_IMAGES	Provide a valid URI for the container image.
INVALID_CONTAINER_IMAGE_URI	The image [invalid_image_uri] doesn't have access to this product. Upload the image to its corresponding product repository. For information about uploading, see Getting started with container products .
INVALID_CONTAINER_IMAGE_TAG	Avoid using 'latest' tag.
MISSING_DELIVERY_OPTION_IDS	Provide delivery option from existing list of Ids.
EMPTY_DELIVERY_OPTION_IDS	Provide non-empty list of delivery option IDs.
DUPLICATE_DELIVERY_OPTION_IDS	Duplicate delivery option IDs: [duplicate_ids]. Provide unique delivery option IDs.
DUPLICATE_DELIVERY_OPTION_TITLES	Duplicate delivery option titles: [duplicate_titles]. Remove the duplicates
INVALID_DELIVERY_OPTION_TITLES	The delivery option titles [existing_titles] already exist. Provide a different title.
INVALID_FULFILLMENT_OPTION_TITLE	Provide delivery option title with fewer than (x) characters.
EMPTY_RESOURCE_NAME	Provide resource name.
EMPTY_RESOURCE_URL	Provide resource URL.
INVALID_RESOURCE_NAME	Provide resource name with fewer than 256 characters.

Error code	Error message
INVALID_RESOURCE_URL	Provide resource URL with fewer than 256 characters.
INVALID_SHORT_DESCRIPTION	Provide a short description with fewer than 1,000 characters.
INVALID_SHORT_DESCRIPTION	Provide short description.
NO_LICENSE_SECRET_KEYS	For Amazon EKS Anywhere products, provide 1 override parameter for license secret. Needs <code>DefaultValue</code> of <code>"\${AWSMP_LICENSE_SECRET}"</code> , see example in section.
NO_SERVICE_ACCOUNT_CONFIGURATION	For paid products, provide 1 override parameter for service account configuration. Needs <code>DefaultValue</code> of <code>"\${AWSMP_SERVICE_ACCOUNT}"</code> , see example in section.
SCAN_ERROR	Fix security vulnerability <code>""[y]""</code> on Image <code>""[x]""</code> .
FIELD_NOT_ALLOWED_TO_CHANGE	Field [x] cannot be changed.
INVALID_DELIVERY_OPTIONS_STATUS	The delivery option IDs [invalid_ids] are invalid. Provide delivery options in the limited or public state.
NO_CHANGE_FOUND	Provide at least 1 change.
MULTIPLE_VERSION_UPDATE	Provide delivery option IDs from the same version.
OVERRIDE_PARAMETER_KEYS_CONTAINS_SPECIAL_CHARS	The override parameter keys [invalid_keys] contain invalid characters. You keys must use only letters, numbers, double quotes (" ") and plus signs (+).

Error code	Error message
INVALID_CONTAINER_IMAGE_REPOSITORY	The repositories [invalid_repositories] are invalid. Provide repositories created through the AddRepositories change type.
INVALID_CONTAINER_IMAGE_TAG_FORMAT	The container image tag [invalid_image_tag] is invalid. Provide a tag conforming to the {CONTAINER_IMAGE_TAG_REGEX} regular expression.
DUPLICATE_OVERRIDE_PARAMETER_KEYS	The override parameters contain duplicate keys [duplicate_keys]. Remove the duplicates.
UNSUPPORTED_CONTAINER_IMAGE_URI	The container image [unsupported_image] is unsupported. Provide an image that follows the https://docs.docker.com/registry/spec/manifest-v2-1/#example-manifest schema.
INVALID_NAMESPACE	The namespace values [invalid_namespaces] are invalid. Provide values that conform to the {HELM_RELEASE_PARAM_REGEX} regular expression.
INVALID_RELEASE_NAME	The releaseName values [invalid_release_names] are invalid. Provide values that conform to the {HELM_RELEASE_PARAM_REGEX} regular expression.
VERRIDE_PARAMETER_KEYS_CONTAINS_RESERVED_PARAMETER_KEYS	The override parameter key for delivery option titles [invalid_keys] is reserved. Reserved keys: [reserved_param_keys]. Provide a different key.

Error code	Error message
INCOMPATIBLE_ADDON_NAME	The add-on name [provided_name] does not match the existing name. Reuse the existing name from the public version, or previous versions, of this add-on. You can only use one add-on name for each product.
INCOMPATIBLE_ADDON_NAMESPACE	The add-on namespace [provided_namespace] does not match the existing namespace . Reuse the existing namespace from the public version, or previous versions, of this add-on. You can only use one add-on namespace for each product.

Restrict a version

You can use the Catalog API to restrict a version of your container-based product in AWS Marketplace. This prevents new buyers from being able to use that version. There must be at least one publicly available version in a product. You cannot restrict the only remaining publicly available version for a product.

To restrict a version, call the `StartChangeSet` API operation with the `RestrictDeliveryOptions` change type, as shown in the following example.

Note

Restricting one or more, but not all, delivery options from a version will remove those options from being available to your buyers. Restricting all delivery options for a version will remove that version from the AWS Marketplace catalog.

Restricting an Amazon EKS add-on is currently not supported through the Catalog API. Restricted versions are still available for existing customers.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
```

```

Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "RestrictDeliveryOptions",
      "Entity":
      {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument":
      {
        "DeliveryOptionIds":
        [
          "example1-2222-cccc-2222-cccccccccccc"
        ]
      }
    }
  ]
}

```

Provide information for the fields to add the RestrictDeliveryOptions change type:

- Entity (object) (required) – Your container-based product.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: ContainerProduct@1.0.
- DetailsDocument (object) (required) – Details of the request. It includes IDs for the delivery options of your container-based product that you would like to restrict.
 - DeliveryOptionIds (array of strings) – List of DeliveryOption IDs for the versions that you want to restrict. You can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the product you are restricting.

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This process can take a few minutes to hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

Asynchronous Errors

The following errors are specific to RestrictDeliveryOptions actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT_STATUS	Use a public product.
MISSING_DELIVERY_OPTION_IDS	Provide delivery option from existing list of IDs.
INVALID_DELIVERY_OPTIONS_STATUS	The delivery option IDs [invalid_ids] are invalid. Provide delivery options in the public state.
EMPTY_DELIVERY_OPTION_IDS	Provide non-empty list of delivery option IDs.
INVALID_MINIMUM_PUBLIC_DELIVERY_OPTIONS	Cannot restrict all delivery option IDs.
DUPLICATE_DELIVERY_OPTION_IDS	Duplicate delivery option IDs: [duplicate_ids]. Provide unique delivery option IDs.

Work with SaaS products using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with SaaS-based products. For information about creating SaaS-based products, see [Create a product](#). The following topics describe how to perform Catalog API actions:

Topics

- [Add delivery options](#)
- [Update delivery options](#)
- [Update delivery option visibility](#)

For information about Quick Launch options, see the following resources:

- (Buyers) For Quick Launch options for SaaS products, see [Configuring and launching SaaS products using Quick Launch](#) in the *AWS Marketplace Buyer Guide*.
- (Sellers) For Quick Launch options for SaaS products, see [Configure Quick Launch](#) in the *AWS Marketplace Seller Guide*.
- For a Quick Launch workshop for SaaS products, see [Lab: Enable SaaS Quick Launch](#) in the *AWS Marketplace seller workshop*.

Add delivery options

You can use the Catalog API to add delivery options for a SaaS product in AWS Marketplace.

To add detailed information for delivery options, call the `StartChangeSet` API operation with the `AddDeliveryOptions` change type to add delivery details, as shown in the following example.

Note

This is only supported for one delivery option: `SaaSUrlDeliveryOptionDetails`. It allows you to add the `FulfillmentUrl` to the SaaS product.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```


```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12@1",
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument": {
        "DeliveryOptions": [
          {
            "Details": {
              "SaaSUrlDeliveryOptionDetails": {
                "FulfillmentUrl": "url to seller's account
registration/login page",
                "QuickLaunchEnabled": true,
                "LaunchUrl": "URL",
                "UsageInstructions": "Instructions",
                "DeploymentTemplates": [
                  {
                    "Title": "CloudFormation Template 123",
                    "Description": "CloudFormation description",
                    "IamPolicy": "{\"Version\":\"2012-10-17\",
\"Statement\": [{\"Effect\":\"Allow\", \"Action\": [\"s3:Get*\", \"s3:List*\"], \"Resource
\": [\"arn:aws:s3:::amzn-s3-demo-bucket\", \"arn:aws:s3:::amzn-s3-demo-bucket/*\"]}]}",
                    "CloudFormationDetails": {
                      "TemplateUrl": "URL",
                      "DefaultStackName": "Name"
                    }
                  }
                ]
              }
            }
          ]
        }
      }
    ]
  }
}

```

Provide information for the fields to add the AddDeliveryOptions change type:

- **Entity (object) (required)** – Your SaaS-based product.
 - **Identifier (string) (required)** – Your product ID. For more information, see [Identifier](#).
 - **Type (string) (required)** – The Type is based on the delivery method (product type) that your product will be using: `SaaSProduct@1.0`.
- **DetailsDocument (object) (required)** – Details of the request.
- **DeliveryOptions (array)** – Details of the delivery options being added.
 - **Details (object)** – Contains the `SaaSUrlDeliveryOptionDetails` of a delivery option to be added.
 - **SaaSUrlDeliveryOptionDetails (object)** – Contains the `FulfillmentUrl` of a delivery option for SaaS product.
 - **FulfillmentUrl (string)** – The URL to be added to the SaaS product.
 - **QuickLaunchEnabled (boolean)** – Determines if buyers can use Quick Launch to configure and launch the software. The default value is `FALSE`.
 - **LaunchUrl (string)** – The URL to your SaaS product's landing page. This is required if `QuickLaunchEnabled` is set to `True`.
 - **DeploymentTemplates (array)** – Deployment templates that customers can use to set up and configure the SaaS product and any related AWS resources.
 - **Title (string)** – The display name of the deployment template.
 - **Description (string)** – A description for what the deployment template contains.
 - **IamPolicy (string)** – An IAM policy describing the permissions needed to deploy the template. Buyers can use this IAM policy to quickly deploy the template.
 - **CloudFormationDetails (object)** – The details of a AWS CloudFormation template.
 - **TemplateUrl (string)** – The URL for the deployment template.
- **DefaultStackName (string)** – The default name used in CloudFormation when the seller creates the stack.

 **Note**

For support obtaining your CloudFormation template URL, contact your AWS Marketplace business development partner or the [AWS Marketplace Seller Operations](#) team.

- `UsageInstructions` (string) – (Optional) Instructions for using this delivery option. Include documentation for manual steps for customers who won't use `DeploymentTemplates`.

Response Syntax

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

Synchronous Validations

Error condition	Message	HTTP code
Invalid Fulfillment Url	Provide a valid fulfillment URL beginning with "https://".	422
Multiple URL Delivery Options	You provided more than one URL delivery option. Provide one URL delivery option.	422
Invalid Delivery Option Types	You provided invalid delivery option details. You must provide a valid parameter for <code>SaaSUrlDeliveryOptionDetails</code> .	422
Invalid Launch URL	Provide a valid launch URL beginning with "https://".	422
Missing Launch Url	Required parameter <code>LaunchUrl</code> is missing. You must provide a <code>LaunchUrl</code> .	422

Error condition	Message	HTTP code
Missing deployment templates	The deployment template is missing. Provide at least one deployment template.	422
Too many deployment templates	You cannot provide more than 20 deployment templates.	422
Invalid template URL	Quick Start URL is invalid. Provide deployment template URL that is published through AWS QuickStarts to Amazon S3. Invalid deployment templates URL: [x]	422
Invalid deployment template stack name	The deployment template stack name is invalid. Provide a valid stack name using only alphanumeric characters and hyphens. It must start with an alphabetical character and can't be longer than 128 characters.	422
Duplicate deployment template title	You provided duplicate deployment template titles. Provide unique deployment template titles.	422
Duplicate deployment template URL	You provided duplicate deployment template urls. Provide unique deployment template urls.	422
Invalid deployment template type	The deployment template type is invalid. Provide a valid deployment template type. Supported values are ["CloudFormation@1.0"].	422

Error condition	Message	HTTP code
Invalid deployment template IAM policy	The deployment template IAM policy is invalid. Provide a valid IAM policy.	422
Invalid usage instructions	<ul style="list-style-type: none"> Images aren't supported by the usage instructions. Remove the image [x]. You provided a link to an invalid URL in the usage instructions: [x]. Provide a valid URL. You provided a link with an unsupported URI scheme in the usage instructions. Use a supported scheme: ["http", "https", "tel", "mailto"]. 	422

Asynchronous Errors

Error code	Error message
DUPLICATE_DELIVERY_OPTIONS	You provided one or more delivery option types that already exist for this product. Provide a unique delivery option type or use <code>UpdateDeliveryOptions</code> if you intended to change an existing delivery option.
INVALID_FULFILLMENT_URL	The URL you provided returned HTTP status code [x]. Provide a fulfillment URL that renders with a 200.
INVALID_LAUNCH_URL	The URL you provided returned HTTP status code [x]. Provide a launch URL that renders with a 200.

Error code	Error message
INVALID_TEMPLATE_URL	Quick Start URL is invalid. Provide deployment template URL that is published through AWS QuickStarts to Amazon S3. Invalid deployment templates URL: [x]

Update delivery options

You can use the Catalog API to update the delivery options for a SaaS product in AWS Marketplace.

To update the delivery options, call the `StartChangeSet` API operation with the `UpdateDeliveryOptions` change type, as shown in the following example.

Note

This is only supported for one delivery option: `SaaSUrlDeliveryOptionDetails`. It allows you to update the `FulfillmentUrl`.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12@1",
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument": {
        "DeliveryOptions": [
          {
            "Id": "do-1234567891234567891234",
            "Details": {
              "SaaSUrlDeliveryOptionDetails": {
```


- `LaunchUrl` (string) – The URL to your SaaS product’s landing page. This is required if `QuickLaunchEnabled` is set to `True`.
- `UsageInstructions` (string) – Instructions for using this delivery option. Include documentation for manual steps for customers who won’t use `DeploymentTemplates`.
- `DeploymentTemplates` (array) – Deployment templates that customers can use to set up and configure the SaaS product and any related AWS resources.
 - `Title` (string) – The display name of the deployment template.
 - `Description` (string) – A description for what the deployment template contains.
 - `IamPolicy` (string) – An IAM policy describing the permissions needed to deploy the template. Buyers can use this IAM policy to quickly deploy the template.
 - `CloudFormationDetails` (object) – The details of a CloudFormation template.
 - `TemplateUrl` (string) – The URL for the deployment template.
 - `DefaultStackName` (string) – The default name used in AWS CloudFormation when the customer creates the template.

Response Syntax

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the `DescribeChangeSet` action.

Synchronous Validations

Error condition	Message	HTTP code
Empty Delivery Option Ids	Provided Details is not valid. String " at /DeliveryOptions/0/Id does not	422

Error condition	Message	HTTP code
	match required schema regex, '^do-[a-zA-Z0-9]+\$'	
Missing Delivery Option Ids	Provided Details is not valid. JSON at /DeliveryOptions/0 is missing required properties: ['Id'].	422
Duplicate Delivery Option Ids	Provide unique delivery option IDs.	422
Invalid Fulfillment Url	Provide a valid fulfillment URL beginning with "https://".	422
Invalid Delivery Option IDs	Provide delivery option IDs that can be found in the product. IDs not found: [x]	422
Multiple URL Delivery Options	You provided more than one URL delivery option. Provide one URL delivery option.	422
Missing delivery option ids	The delivery option ID is missing. Provide one or more valid delivery option IDs that you wish to update, or use <code>AddDeliveryOptions</code> if you intended to add a new delivery option.	422
Invalid Launch URL	Provide a valid launch URL beginning with "https://".	422
Missing Launch Url	Required parameter <code>LaunchUrl</code> is missing. You must provide a <code>LaunchUrl</code> .	422
Missing deployment templates	The deployment template is missing. Provide at least one deployment template.	422

Error condition	Message	HTTP code
Too many deployment templates	You cannot provide more than 20 deployment templates.	422
Invalid template URL	Quick Start URL is invalid. Provide deployment template URL that is published through AWS QuickStarts to Amazon S3. Invalid deployment templates URL: [x]	422
Invalid deployment template stack name	The deployment template stack name is invalid. Provide a valid stack name using only alphanumeric characters and hyphens. It must start with an alphabetical character and can't be longer than 128 characters.	422
Duplicate deployment template title	You provided duplicate deployment template titles. Provide unique deployment template titles.	422
Duplicate deployment template URL	You provided duplicate deployment template urls. Provide unique deployment template urls.	422
Invalid deployment template type	The deployment template type is invalid. Provide a valid deployment template type. Supported values are ["CloudFormation@1.0"].	422
Invalid deployment template IAM policy	The deployment template IAM policy is invalid. Provide a valid IAM policy.	422

Error condition	Message	HTTP code
Invalid usage instructions	<ul style="list-style-type: none"> Images aren't supported by the usage instructions. Remove the image [x]. You provided a link to an invalid URL in the usage instructions: [x]. Provide a valid URL. You provided a link with an unsupported URI scheme in the usage instructions. Use a supported scheme: ["http", "https", "tel", "mailto"]. 	422

Asynchronous Errors

Error code	Error message
INVALID_DELIVERY_OPTION_IDS	Provide delivery option IDs that can be found in the product. IDs not found: [x]
AUDIT_ERROR	AWS MP Catalog Audits List - CQ team
INVALID_FULFILLMENT_URL	The URL you provided returned HTTP status code [x]. Provide a fulfillment URL that renders with a 200.
INVALID_LAUNCH_URL	The URL you provided returned HTTP status code [x]. Provide a launch URL that renders with a 200.
INVALID_TEMPLATE_URL	Quick Start URL is invalid. Provide deployment template URL that is published through AWS QuickStarts to Amazon S3. Invalid deployment templates URL: [x]

Update delivery option visibility

You can use the Catalog API to configure permissions so that only some users can change the visibility for a SaaS product in AWS Marketplace.

To configure permissions so that only some users can change the visibility for a SaaS product, call the `StartChangeSet` API operation with the `UpdateDeliveryOptionsVisibility` change type, as shown in the following example.

Note

This is only supported for one delivery option: `SaaSUrlDeliveryOptionDetails`.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateDeliveryOptionsVisibility",
      "Entity":
      {
        "Identifier": "prod-example12345",
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument":
      {
        "DeliveryOptions":
        [
          {
            "Id": "do-1234567891234567891234",
            "TargetVisibility": "Public"
          },
          {
            "Id": "do-43210987654321",
            "TargetVisibility": "Limited",
            "Targeting":

```



```

    {
      "PositiveTargeting":
        {
          "BuyerAccounts":
            [
              "123456789012"
            ]
          }
        }
      ]
    }
  ]
}

```

Provide information for the fields to add the UpdateDeliveryOptionsVisibility change type:

- Entity (object) (required) – Your SaaS-based product.
 - Identifier (string) (required) – Your product ID. For more information, see [Identifier](#).
 - Type (string) (required) – The Type is based on the delivery method (product type) that your product will be using: SaaSProduct@1.0.
- DetailsDocument (object) (required) – Details of the request.
 - DeliveryOptions (array) – List of DeliveryOptions to be updated.
 - TargetVisibility (string) – The delivery option id to be updated.
 - TargetVisibility (string) – The intended new visibility of the delivery option.

Possible values: Limited, Public, and Unavailable.

Note

There is always exactly one Public delivery option, and a maximum of one Limited delivery option.

- Targeting (object) *optional* – Targeting of the delivery option, used in conjunction with the Limited visibility status to be able to test the new delivery option before changing the visibility to Public.
 - PositiveTargeting (object) – Specifying inclusive targeting.

- `BuyerAccounts` (array of strings) – The list of buyer AWS account ids who will be able to use the new delivery option.

Min size: 0. Max size: 100.

Response Syntax

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the `DescribeChangeSet` action.

Synchronous Validations

Error condition	Message	HTTP code
Missing delivery option ids	The delivery option ID is missing. Provide one or more valid delivery option IDs that you wish to update, or use <code>AddDeliveryOptions</code> if you intended to add a new delivery option.	422
Invalid visibility	You provided invalid option for <code>TargetVisibility</code> . Allowed options are: <code>Limited</code> , <code>Public</code> , <code>Unavailable</code> .	422
Invalid targeting	You provided invalid option for <code>PositiveTargeting</code> . You must provide a valid parameter for <code>BuyerAccounts</code> .	422

Error condition	Message	HTTP code
Missing Visibility and Targeting	You provided invalid delivery option visibility details. You must provide a valid parameter for at least one of <code>TargetVisibility</code> or <code>Targeting</code> .	422
Too many AWS account ids	You cannot provide more than 100 targeted buyer accounts.	422

Asynchronous Errors

Error code	Error message
INVALID_DELIVERY_OPTION_IDS	You provided invalid delivery option details. Provide delivery option IDs that can be found in the product. IDs not found: [x]
INVALID_VISIBILITY	You provided more than one delivery option for the public state. Provide only one public delivery option.
INVALID_VISIBILITY	You didn't provide a public delivery option. Provide one public delivery option.
AUDIT_ERROR	Varies based on MCO manual review.

Work with offers using the AWS Marketplace APIs

An offer defines the terms and rules governing the purchase and consumption of a product. It encompasses a collection of agreement terms between two parties. The accepted offer terms are then translated into an agreement, serving as proof of the transaction.

The following sections provide information on how to manage offers as a seller or channel partner.

Topics

- [Work with private offers using the AWS Marketplace APIs](#)
- [Work with resale authorizations using the AWS Marketplace APIs](#)
- [Work with channel partner private offers using the AWS Marketplace APIs](#)
- [Work with renewals](#)

Work with private offers using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with offers.

While the *product* describes what is being sold in AWS Marketplace, the *offer* describes the terms and rules of how a product is purchased and consumed. AWS Marketplace products can have multiple offers sold by different sellers. Each AWS Marketplace offer, however, can only be created for one product. An *offer* contains a collection of agreement terms between two parties. The accepted offer terms are translated into an agreement as proof of a transaction.

There are two types of offers:

- **Private offers** are for sellers and buyers to negotiate pricing. Sellers sign an end-user license agreement (EULA) for software purchases in AWS Marketplace. An offer is visible only to a specified buyer. For more information, see [Private offers](#) in the *AWS Marketplace Seller Guide*.
- **Public offers** are for global purchasing programs. Sellers identify customers based on available programs and geographical locations, which makes the offer accessible only to specific customers.

See the following resources:

- For working code examples, see [Manage offers with API](#) in the *AWS Marketplace seller workshop*.
- For API request code examples, see [Python](#) and [Java](#) examples in *AWS Samples* on GitHub.
- For a video on creating private offers, see [Create a Private Offer Using the AWS Marketplace Catalog API](#) on YouTube.
- For a video on updating AMI pricing, see [Update AMI Product Pricing Using the AWS Marketplace Catalog API](#) on YouTube.

The following topics describe how to use the Catalog API to create and update offers:

Topics

- [Create an offer](#)
- [Create a replacement offer](#)
- [Update offer information](#)
- [Update targeting configuration](#)
- [Update refund policy](#)
- [Update legal resources](#)
- [Update pricing](#)
- [Update the discoverability of the offer](#)
- [Define the expiration date of agreements created using the offer](#)
- [Update payment schedule details](#)
- [Modify renewal options](#)
- [Publish an offer](#)
- [Describe existing offer details](#)

Create an offer

You can use the Catalog API to create a new offer in AWS Marketplace. If your request processes successfully, the AWS Marketplace Catalog API creates a `Draft`, which is an incomplete offer that's invisible to buyers. To complete an offer, use the `Update` change type. When the offer is complete, use the [ReleaseOffer](#) change type to create and release it. Releasing an offer validates it and makes it visible to buyers in AWS Marketplace.

To create a new offer, call the `StartChangeSet` API operation with the `CreateOffer` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
```

```
    },
    "DetailsDocument": {
      "ProductId": "prod-ad8EXAMPLE51",
      "Name": "Test Offer"
    }
  }
]
```

Provide information for the fields to add the `CreateOffer` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **ProductId** (string) (required) – The unique identifier of the product being offered.
 - **Name** (string) (optional) – The name associated with the offer for better readability to you and your customers. It is displayed as a part of the Agreement information as well.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

If the `Status` is `SUCCEEDED`, then a new `OfferId` is generated.

The response looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
```

```

"ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef",
"ChangeSetName": "Submitted by 123456789012",
"StartTime": "2021-05-27T22:21:26Z",
"EndTime": "2021-05-27T22:32:19Z",
"Status": "SUCCEEDED",
"ChangeSet": [
  {
    "ChangeType": "CreateOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "offer-123456789"
    },
    "DetailsDocument": {
      "ProductId": "prod-ad8EXAMPLE51",
      "Name": "Test Offer"
    },
    "ErrorDetailList": []
  }
]
}

```

Synchronous Validations

The following schema validations are specific to `CreateOffer` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
ProductId	Required Length must be between 1 and 50 characters Must not contain illegal characters (\, <, >)	422
ProductId	Required User must be authorized to create offer for the given product	403

Input field	Validation rule	HTTP code
ProductId	Required Must be an existing product in the catalog or being created in the same change set	404
Name	Optional Length must be between 1 and 150 characters Must not contain illegal characters (\, <, >)	422

Asynchronous Errors

The following errors are specific to `CreateOffer` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT	Use an active product in <code>Limited</code> or <code>Public</code> state.
INCOMPATIBLE_PRODUCT	Managing offers for your chosen product type isn't currently supported in the AWS Marketplace Catalog API.

Create a replacement offer

You can use the Catalog API to create a replacement offer (also known as an agreement-based offer) in AWS Marketplace.

If your request has been processed successfully, AWS Marketplace Catalog API will have an offer in `Draft` state generated for you, which is an incomplete offer and not visible to buyers on AWS

Marketplace. You will use Update change types to complete the offer. After the offer is completed, you will use [ReleaseOffer](#) change type to complete offer creation process and release the offer, which will validate the entire offer and make your offer visible to buyers on AWS Marketplace. From there, the buyer has the option to accept the replacement offer or to continue to operate under the original agreement.

To create a replacement offer, call the StartChangeSet API operation with the CreateReplacementOffer change type and provide a pre-existing agreement id, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateReplacementOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "AgreementId": "agmt-12345",
        "Name": "Offer name"
      }
    }
  ]
}
```

Provide information for the fields to add the CreateReplacementOffer change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always Offer@1.0.
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **AgreementId** (string) (required) – The unique identifier for the current agreement to be replaced.
 - **Name** (string) (optional) – The name associated with the offer for better readability to you and your customers. It will be displayed as part of Agreement information as well.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `CreateReplacementOffer` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP Code
AgreementId	Required Length must be between 1 and 64 characters	422
AgreementId	Required User must be authorized to create offer for the given agreement	403
Name	Optional Length must be between 1 and 150 characters Must not contain invalid characters (\, <, >)	422

Asynchronous Errors

The following errors are specific to `CreateReplacementOffer` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT	Replacement offers aren't supported for the product.

Update offer information

You can use the Catalog API to update the offer information in AWS Marketplace.

To update the offer information, call the `StartChangeSet` API operation with the `UpdateInformation` change type, as shown in the following example. All other information will remain unchanged.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Name": "New offer name",
        "Description": "New offer description",
        "PreExistingAgreement": {
          "AcquisitionChannel": "External",
          "PricingModel": "Contract"
        }
      }
    }
  ]
}
```

```

    }
  }
}
]
}

```

Provide information for the fields to add the `UpdateInformation` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Name** (string) (optional) – Name associated with the offer for better readability. It is displayed as part of agreement information.
 - **Description** (string) (optional) – A free-form text that is meant to be used only by you and will never be visible to buyers.
 - **PreExistingAgreement** (object) (optional) – Determines if this offer is a renewal for an existing agreement with an existing customer for the same underlying product. The existing agreement can be within or outside AWS Marketplace. AWS may audit and verify your offer is a renewal. If AWS is unable to verify your offer, then AWS may revoke the offer and entitlements from your customer.
 - **AcquisitionChannel** (string) (required) – Indicates if the existing agreement was signed outside AWS Marketplace or within AWS Marketplace.

Possible values: `External`, `AwsMarketplace`

- **PricingModel** (string) (required) – Indicates which pricing model the existing agreement uses.

Possible values: `Contract`, `Usage`, `ByoI`, `Free`

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```

{
  "ChangeSetId": "example123456789012abcdef",

```

```
"ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Properties	At least one of the following properties must be provided	422
Name	Optional Length must be between 1 and 150 characters Must not contain illegal characters (\, <, >)	422
Description	Optional Length must be between 1 and 255 characters	422
PreExistingAgreement	Optional Can be null to remove PreExistingAgreement from offer	422
PreExistingAgreement.PricingModel	Required	422

Input field	Validation rule	HTTP code
	Can be one of these values: [Byo], Free, Usage, Contract]	
PreExistingAgreement.AcquisitionChannel	Required Can be one of these values: [AwsMarketplace , External]	422

Asynchronous Errors

The following errors are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRE_EXISTING_AGREEMENT	PreExistingAgreement can't be changed after the offer is released.

Update targeting configuration

You can use the Catalog API to update the targeting configuration of your offer in AWS Marketplace.

All existing targeting options that aren't included in the latest request and will be removed from the offer.

To update the targeting configuration of your offer, call the StartChangeSet API operation with the UpdateTargeting change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "CountryCodes": [
            "US",
            "CA"
          ],
          "BuyerAccounts": [
            "111122223333"
          ]
        },
        "NegativeTargeting": {
          "CountryCodes": [
            "XX"
          ]
        }
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateTargeting change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always Offer@1.0.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **PositiveTargeting** (object) (optional) – Positive targeting defines the criteria which any buyer's profile should fulfill in order to be allowed to access the offer. This field is optional, but at least one targeting option should be provided when this field is present.
 - **CountryCodes** (array of strings) (optional) – List as option for allowing targeting based on country. If the intention isn't to target the offer to a country, this field should be omitted. If

it's present, the list must contain at least one country code. Each element in this list should be a valid 2-letter country code, using this format: ISO 3166-1 alpha-2.

- **BuyerAccounts** (array of strings) (optional) – List as an option to allow targeting based on AWS accounts (also known as Private Offer). If the intention is to not target the offer to an AWS account, this field should be omitted.
- **NegativeTargeting** (object) (optional) – Negative targeting defines the criteria which any customer's profile should fulfill to be restricted to access the offer. Although this field is optional, at least one targeting option should be provided when this field is present.
- **CountryCodes** (array of strings) (required) – List as option for allowing targeting based on country. If the intention isn't to target the offer to a specific country, then this field should be omitted. If it's present, the list must contain at least one country code. Each element in this list should be a valid 2-letter country code using this format: ISO 3166-1 alpha-2.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateTargeting` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
NegativeTargeting	Optional Can have either one of the following : [CountryCodes]	422
NegativeTargeting.CountryCodes	Optional List size must be between 1 and 244 Country codes must be valid (ISO 3166-1 alpha-2)	422
PositiveTargeting	Optional Can have either one of the following : [CountryCodes , BuyerAccounts]	422
PositiveTargeting.BuyerAccounts	Optional List size must be between 1 and 26 AWS account IDs must be in valid format (12-digit number)	422
PositiveTargeting.CountryCodes	Optional List size must be between 1 and 244 Country codes must be valid (ISO 3166-1 alpha-2)	422

Asynchronous Errors

The following errors are specific to UpdateTargeting actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_BUYER_ACCOUNTS	Provide valid buyer accounts. Invalid accounts: [x].
INVALID_COUNTRY_CODES	Provide supported country codes.
INVALID_TARGETING	Use either negative or positive targeting on the same attribute.
INCOMPATIBLE_PRODUCT	Country-based targeting isn't supported for the product.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide BuyerAccounts that are compatible with the ResaleAuthorization.
INCOMPATIBLE_TARGETING	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TARGETING	The requested change can't be performed after the offer is expired.
INCOMPATIBLE_TARGETING	Targeting can't be updated on a replacement offer. If the buyer isn't associated with the provided AgreementId, then create a new private offer by providing an AgreementId associated with the buyer.
TOO_MANY_BUYER_ACCOUNTS	Provide BuyerAccounts within the allowed limits.

Update refund policy

You can use the Catalog API to update the refund policy of your offer in AWS Marketplace.

This change doesn't affect existing agreements. The support terms that aren't included in the latest request will be removed from the offer.

To update the refund policy, call the `StartChangeSet` API operation with the `UpdateSupportTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateSupportTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "SupportTerm",
            "RefundPolicy": "Updated refund policy description"
          }
        ]
      }
    }
  ]
}
```

Provide information for the fields to add the `UpdateSupportTerms` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Terms** (array of structures) (required) – List of support terms that you would like to update. Accepted support terms are:
 - **SupportTerm** (object) (required) – Defines the customer support available for the acceptors when they purchase the software.

- **Type** (string) (required) – Type of the term being updated. This is the object value: "SupportTerm".
- **RefundPolicy** (string) (required) – Free-text field about the refund policy description that will be shown to customers as is on the website and console.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateSupportTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required	422
Terms[].RefundPolicy	Required Length must be between 1 and 500 Cannot lead or end with spaces	422
Terms[].Type	Required	422

Input field	Validation rule	HTTP code
	Can only be SupportTerm	

Asynchronous Errors

The following errors are specific to UpdateSupportTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT	SupportTerm isn't supported in private offers for the product.
INCOMPATIBLE_TERMS	SupportTerm isn't supported for free trial offers.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.

Update legal resources

You can use the Catalog API to replace the existing legal documents, such as an end user license agreement (EULA). The legal terms that aren't included in the latest request will be removed from the offer.

To update legal resources of your offer, call the StartChangeSet API operation with the UpdateLegalTerms change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "LegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

Provide information for the fields to add the `UpdateLegalTerms` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **LegalTerm** (object) (required) – Defines the list of text agreements to be proposed to the acceptors. One example of such an agreement is the end user license agreement (EULA).
 - **Type** (string) (required) – Type of the term being updated. This is the object value: `"LegalTerm"`.
 - **Documents** (array of structures) (required) – List of references to legal resources to be proposed to the buyers. One example of such a resource is the end user license agreement (EULA). Each reference is made up of a Type and a URL:

- **Type** (string) (required) – Type of document. Available document types are:
 - **CustomEula** – A custom EULA provided by you as seller. A URL for a EULA stored in an accessible S3 bucket is required for this document type.
 - **StandardEula** – Standard Contract For AWS Marketplace (SCMP). For more information about SCMP, see the AWS Marketplace Seller Guide. You don't provide a URL for this type because it is managed by AWS Marketplace.
- **Url** (string) (conditionally required) – A URL to the legal document for buyers to read. Required when Type is one of the following [CustomEula].
- **Version** (string) (conditionally required) – Version of standard contracts provided by AWS Marketplace. Required when Type is [StandardEula]. Available version:
 - **2022-07-14** – This version of the Standard Contract for AWS Marketplace is available from this Amazon S3 bucket: <https://s3.amazonaws.com/aws-mp-standard-contracts/Standard-Contract-for-AWS-Marketplace-2022-07-14.pdf>

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateLegalTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required Only LegalTerm is allowed in the list List size must be 1	422
Terms[].Type	Required Can only be LegalTerm	422
Terms[].LegalTerm.Documents	Required	422
Terms[].LegalTerm.Documents[].Type	Required Allowed values: <ul style="list-style-type: none">• CustomEula• StandardEula	422
Terms[].LegalTerm.Documents[].Url	Required and must be a valid URL when Type is CustomEula	422
Terms[].LegalTerm.Documents[].Version	Required and must be a valid Version when Type is StandardEula Valid StandardEula versions: ["2019-04-24", "2022-07-14"]	422

Asynchronous Errors

The following errors are specific to UpdateLegalTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_LEGAL_DOCUMENTS	Provide URLs for legal documents stored in accessible S3 buckets.
INVALID_LEGAL_DOCUMENTS	Only the most recent version of StandardEula is supported for new offers.
INVALID_LEGAL_DOCUMENTS	Provide legal documents in the supported file formats.
INVALID_LEGAL_DOCUMENTS	Provide legal documents using the supported document types.
LIMIT_EXCEEDED_LEGAL_DOCUMENT_SIZE	Provide legal documents within the allowed size limits.

Update pricing

You can use the Catalog API to replace the existing pricing terms completely. The pricing terms that aren't included in the latest request will be removed from the offer.

To update pricing terms for your offer, call the `StartChangeSet` API operation with the `UpdatePricingTerms` change type, as shown in the following example.

Note

The following request syntax combines multiple examples. This combination doesn't work as a valid payload. For example, a `Terms` array can't include both the term type `FixedUpfrontPricingTerm` and the term type `ConfigurableUpfrontPricingTerm`. For examples of how different term types are combined for different pricing use cases, see [Manage offers with API](#) in the *AWS Marketplace seller workshop*.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
          {
            "Type": "UsageBasedPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "RateCard": [
                  {
                    "DimensionKey": "m3.large",
                    "Price": "0.10"
                  },
                  {
                    "DimensionKey": "m4.xlarge",
                    "Price": "0.20"
                  }
                ]
              }
            ]
          },
          {
            "Type": "ConfigurableUpfrontPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "Selector": {
                  "Type": "Duration",
                  "Value": "P365D"
                }
              }
            ]
          }
        ]
      }
    }
  ]
}
```

```
    "RateCard": [
      {
        "DimensionKey": "m3.large",
        "Price": "300"
      },
      {
        "DimensionKey": "m4.xlarge",
        "Price": "400"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
],
{
  "Type": "ByolPricingTerm"
},
{
  "Type": "RecurringPaymentTerm",
  "CurrencyCode": "USD",
  "BillingPeriod": "Monthly",
  "Price": "100.0"
},
{
  "Type": "FixedUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "Price": "200.00",
  "Grants": [
    {
      "DimensionKey": "Users",
      "MaxQuantity": 10
    }
  ]
},
{
  "Type": "FreeTrialPricingTerm",
  "Duration": "P30D",
  "Grants": [
    {
      "DimensionKey": "m3.xlarge",
      "MaxQuantity": 10
    }
  ]
}
```


- **Duration** (string) – Duration of the free trial period.
- **Grants** (array of structures) – Entitlements that will be granted to the acceptor of a free trial as part of an agreement execution.
 - **DimensionKey** (string) – Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
 - **MaxQuantity** (integer) (optional) – Maximum amount of capacity that the buyer can be entitled to the given dimension of the product. If `MaxQuantity` is not provided, the buyer will be able to use an unlimited amount of the given dimension.
- **UsageBasedPricingTerm** (object) – Defines a pay-as-you-go (PAYG) pricing model where the customers are charged based on product usage.
 - **Type** (string) (required) – Category of the term being updated. This is the object value: `UsageBasedPricingTerm`.
 - **CurrencyCode** (string) – Defines the currency for prices mentioned in this term. Currently, only USD is supported.
 - **RateCards** (array of structures) – List of rate cards.
 - **RateCard** (array of structures) – A rate card defines the per-unit rates for the product dimensions.
 - **DimensionKey** (string) – Dimension that the given entitlement applies. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
 - **Price** (string) – Per-unit price for the product dimension that will be used for calculating the amount to be charged to the buyer.
- **ConfigurableUpfrontPricingTerm** (object) – Defines pre-paid payment model which allows buyers to configure the entitlements that they want to purchase and the duration of the entitlements. You can update the list of rates for each contract duration and entitlements for each dimension.
 - **Type** (string) (required) – Type of the term being updated. This is the object value: `ConfigurableUpfrontPricingTerm`.
 - **CurrencyCode** (string) (required) – Defines the currency for the prices mentioned in this term. For public offers, only USD is supported. For private offers, USD, AUD, EUR, GBP, and JPY are supported.
 - **RateCards** (array of structures) (required) – List of rate cards.

- **Selector** (object) (required) – Selector is used to differentiate between the mutually exclusive rate cards in the same pricing term, to be selected by the buyer.
- **Type** (string) (required) – Category of Selector. At this time, only `Duration` is supported.
- **Value** (string) (required) – Contract duration. This field supports the ISO 8601 format.
- **RateCard** (array of structures) (required) – A rate card defines the per-unit rates for the product dimensions.
 - **DimensionKey** (string) (required) – Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
 - **Price** (string) (required) – Per-unit price for the product dimension which will be used for calculating the amount to be charged to the buyer.
- **Constraints** (object) (required) – Defines constraints on how the term can be configured by acceptors.

 **Note**

Currently, **MultipleDimensionSelection** and **QuantityConfiguration** values need to be same.

- **MultipleDimensionSelection** (string) (required) – Determines if buyers are allowed to select multiple dimensions in the rate card. Possible values are `Allowed` and `Disallowed`.
- **QuantityConfiguration** (string) (required) – Determines if acceptors are allowed to configure quantity for each dimension in rate card. Possible values are `Allowed` and `Disallowed`.
- **ByolPricingTerm** (object) – Enables you and your customers to move your existing agreements to AWS Marketplace. The customer won't be charged for product usage in AWS Marketplace because they already paid for the product outside of AWS Marketplace.
 - **Type** (string) (required) – Type of the term being updated. This is the object value: `ByolPricingTerm`.
- **RecurringPaymentTerm** (object) – Defines a pricing model where customers are charged a fixed recurring price at the end of each billing period.

- **Type** (string) (required) – Type of the term being updated. This is the object value: `RecurringPaymentTerm`.
- **BillingPeriod** (string) (required) – Defines the recurrence at which buyers are charged. Only `Monthly` is supported today.
- **Price** (string) (required) – Amount charged to the buyer every billing period.
- **CurrencyCode** (string) (required) – Defines the currency for the prices mentioned in this term. Currently, only USD is supported.
- **FixedUpfrontPricingTerm** (object) – Defines a pre-paid pricing model where the customers are charged a fixed upfront amount.
 - **Type** (string) (required) – Type of the term being updated. This is the object value: `FixedUpfrontPricingTerm`.
 - **CurrencyCode** (string) (required) – Defines the currency for the prices mentioned in this term. For public offers, only USD is supported. For private offers, USD, AUD, EUR, GBP, and JPY are supported.
 - **Price** (string) (required) – Fixed amount to be charged to the customer when this term is accepted.
 - **Grants** (array of structures) (required) – Entitlements that will be granted to the acceptor of fixed upfront as part of agreement execution.
 - **DimensionKey** (string) (required) – Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
 - **MaxQuantity** (integer) (required) – Maximum amount of capacity that the buyer can be entitled to the given dimension of the product. If `MaxQuantity` is not provided, the buyer will be able to use an unlimited amount of the given dimension.
 - **Duration** (string) (optional) – Defines the duration that the term remains active. This field supports the ISO 8601 format.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{  
  "ChangeSetId": "example123456789012abcdef",
```

```
"ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdatePricingTerms actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP
PricingModel	Required Allowed pricing models: ["Byol", "Free", "Usage", "Contract"]	422
Terms	Required Allowed Terms: ["ConfigurableUpfrontPricingTerm", "ByolPricingTerm", "FreeTrialPricingTerm", "UsageBasedPricingTerm", "RecurringPaymentTerm", "FixedUpfrontPricingTerm"]	422
Terms[].ByolPricingTerm	Required	422
Terms[].ByolPricingTerm.Type	Required Can only be "ByolPricingTerm"	422
Terms[].ConfigurableUpfrontPricingTerm	Required	422

Input field	Validation rule	HTTP
Terms[].ConfigurableUpfrontPricingTerm.Type	Required Can only be "ConfigurableUpfrontPricingTerm"	422
Terms[].ConfigurableUpfrontPricingTerm.CurrencyCode	Required Supported currencies: ["USD", "AUD", "EUR", "GBP", "JPN"]	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards	Required List size must be between 1 and 5	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Constraints	Required	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Constraints.MultipleDimensionSelection	Required Allowed values: ["Allowed", "Disallowed"]	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Constraints.QuantityConfiguration	Required Allowed values: ["Allowed", "Disallowed"]	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].RateCard	Required List size must be between 1 and 800	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].RateCard[].DimensionKey	Required Length must be between 1 and 100	422

Input field	Validation rule	HTTP
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].RateCard[].Price	Required Data type is "String" Non-negative decimals with up to 3 decimal places supported	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Selector	Required	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Selector.Type	Required Allowed values: ["Duration"]	422
Terms[].ConfigurableUpfrontPricingTerm.RateCards[].Selector.Value	Required Expected format per Selector type: ISO 8601 duration	422
Terms[].FixedUpfrontPricingTerm	Required	422
Terms[].FixedUpfrontPricingTerm.Type	Required Can only be "FixedUpfrontPricingTerm"	422
Terms[].FixedUpfrontPricingTerm.CurrencyCode	Required Supported currencies: ["USD", "AUD", "EUR", "GBP", "JPN"]	422
Terms[].FixedUpfrontPricingTerm.Duration	Required Expected format per Selector type: ISO 8601 duration	422
Terms[].FixedUpfrontPricingTerm.Grants	Required List size must be between 1 and 200	422

Input field	Validation rule	HTTP
Terms[].FixedUpfrontPricingTerm.Grants[].DimensionKey	Required Length must be between 1 and 100	422
Terms[].FixedUpfrontPricingTerm.Grants[].MaxQuantity	RequiredValue must be greater than 0	422
Terms[].FixedUpfrontPricingTerm.Price	Required Data type is "String" Non-negative decimals with up to 3 decimal places supported	422
Terms[].FreeTrialPricingTerm	Required	422
Terms[].FreeTrialPricingTerm.Type	Required Can only be "FreeTrialPricingTerm"	422
Terms[].FreeTrialPricingTerm.Duration	Required Expected format: ISO 8601 duration	422
Terms[].FreeTrialPricingTerm.Grants	Required List size must be between 1 and 800	422
Terms[].FreeTrialPricingTerm.Grants[].DimensionKey	Required Length must be between 1 and 100	422
Terms[].FreeTrialPricingTerm.Grants[].MaxQuantity	Optional Value must be greater than 0	422
Terms[].RecurringPaymentTerm	Required	422

Input field	Validation rule	HTTP
Terms[].RecurringPaymentTerm.Type	Required Can only be "RecurringPaymentTerm"	422
Terms[].RecurringPaymentTerm.BillingPeriod	Required Allowed values: ["Monthly"]	422
Terms[].RecurringPaymentTerm.CurrencyCode	Required Supported currencies: ["USD"]	422
Terms[].RecurringPaymentTerm.Price	Required Data type is "String" Non-negative decimals with up to 3 decimal places supported	422
Terms[].UsageBasedPricingTerm	Required	422
Terms[].UsageBasedPricingTerm.Type	Required Can only be "UsageBasedPricingTerm"	422
Terms[].UsageBasedPricingTerm.CurrencyCode	Required Supported currencies: ["USD"]	422
Terms[].UsageBasedPricingTerm.RateCards	Required Must be size of 1	422
Terms[].UsageBasedPricingTerm.RateCards[].RateCard	Required List size must be between 1 and 800	422

Input field	Validation rule	HTTP
Terms[].UsageBasedPricingTerm.RateCards[].RateCard[].DimensionKey	Required Length must be between 1 and 100	422
Terms[].UsageBasedPricingTerm.RateCards[].RateCard[].Price	Required Data type is "String" Non-negative decimals with up to 8 decimal places supported	422

Asynchronous Errors

The following errors are specific to UpdatePricingTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
DUPLICATE_DIMENSION_KEYS	Provide Grants with a unique list of dimension keys in [x].
DUPLICATE_DIMENSION_KEYS	Provide RateCard with a unique list of dimension keys in [x].
DUPLICATE_SELECTORS	Provide a unique list of Selectors in ConfigurableUpfrontPricingTerm.
DUPLICATE_TERM_TYPES	Provide a unique list of term types.
INCOMPATIBLE_AGREEMENT	The following terms can't be removed from the replacement offer: [x, y, z].
INCOMPATIBLE_AGREEMENT	The following terms can't be added to the replacement offer: [x, y, z].

Error code	Error message
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_PRODUCT	Usage pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	Contract pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	Byol pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	Free pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	[x] isn't supported in an offer for the product.
INCOMPATIBLE_PRODUCT	Provided payment and pricing terms are incompatible.
INCOMPATIBLE_PRODUCT	Use existing, available dimensions in the product in [x].
INCOMPATIBLE_PRODUCT	FreeTrialPricingTerm as the offer's only pricing term isn't supported for the product.
INCOMPATIBLE_PRODUCT	The following terms aren't supported for the product: [x,y,z].
INCOMPATIBLE_PRODUCT	Replacement offers are only supported for contract pricing model.
INCOMPATIBLE_PRODUCT	Provide pricing term(s) that are compatible with the product dimensions. Incompatible pricing terms: [x,y,z].

Error code	Error message
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	Set MultipleDimensionSelection and QuantityConfiguration to Allowed in ConfigurableUpfrontPricingTerm for usage pricing model.
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	Set MultipleDimensionSelection and QuantityConfiguration to Disallowed in ConfigurableUpfrontPricingTerm for usage pricing model.
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	QuantityConfiguration in ConfigurableUpfrontPricingTerm can't be changed after the offer is released.
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	MultipleDimensionSelection in ConfigurableUpfrontPricingTerm can't be changed after the offer is released.
INCOMPATIBLE_RATES	Set all charge amounts and prices to zero (0) when using Free pricing model.
INCOMPATIBLE_RATES	Only zero (0) prices are allowed in UsageBasedPricingTerm for a free trial offer for the product.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide the same CurrencyCode that is specified in the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure Duration in FixedUpfrontPricingTerm matches duration specified in the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide term(s) that are compatible with the ResaleAuthorization. Incompatible terms: [x, y, z].

Error code	Error message
INCOMPATIBLE_SELECTOR_DURATION	Durations aren't allowed to be removed from rate cards in ConfigurableUpfrontPricingTerm after the offer released.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INCOMPATIBLE_TERMS	[x] isn't supported together with the following terms: [y,z].
INCOMPATIBLE_TERMS	The following terms can't be added after the offer is released: [x,y,z].
INCOMPATIBLE_TERMS	The following terms can't be removed after the offer is released: [x,y,z].
INCOMPATIBLE_TERMS	[x] isn't supported for private offers.
INCOMPATIBLE_TERMS	The following terms aren't supported with FreeTrialPricingTerm that grants unlimited usage: [x,y,z].
INCOMPATIBLE_TERMS	The following terms aren't supported with FreeTrialPricingTerm for the product: [x,y,z].
INCOMPATIBLE_TERMS	Provide zero (0) price for FixedUpfrontPricingTerm when the offer contains a PaymentScheduleTerm.
INCOMPATIBLE_TERMS	The following terms aren't compatible with the PricingModel: [x,y,z].
INCOMPATIBLE_TERMS	FixedUpfrontPricingTerm isn't supported when MarkupPercentage is greater than zero (0).
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.

Error code	Error message
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_AGREEMENT_DURATION	Provide duration between [x] and [y] months.
INVALID_AGREEMENT_DURATION	Ensure duration granularity is at the day level for metered dimensions.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INVALID_DURATION	Ensure Duration in FreeTrialPricingTerm is within the allowed range.
INVALID_DURATION	Provide Duration in FixedUpfrontPricingTerm that matches the duration between AgreementStartDate and AgreementEndDate.
INVALID_DURATION	Provide duration between [x] and [y] months.
INVALID_DURATION	Ensure duration granularity is at the day level for metered dimensions.
INVALID_GRANTS	Provide the same MaxQuantity for all Grants in FreeTrialPricingTerm.
INVALID_GRANTS	Provide Grants for all available metered dimensions in FreeTrialPricingTerm.

Error code	Error message
INVALID_PRICE_CHANGE	[x] can't be updated until [y] because you have requested a price increase in the past 120 days. To cancel your previous price increase request or for more information, contact the AWS Marketplace Managed Catalog Operations Team.
INVALID_PRICE_CHANGE	Price increase and dimension addition in [x] isn't supported in the same request. Add dimensions first.
INVALID_PRICE_CHANGE	Price increase and decrease in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.
INVALID_PRICE_CHANGE	Price increase in RecurringPaymentTerm and price decrease in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.
INVALID_PRICE_CHANGE	Price decrease in RecurringPaymentTerm and price increase in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.
INVALID_RATE_CARD	ConfigurableUpfrontPricingTerm is missing one or more-dimension keys for duration [x]. Provide prices for the same set of dimension keys for all durations.
INVALID_RATE_CARD	Provide a rate card for only metered dimensions in UsageBasedPricingTerm.
INVALID_RATE_CARD	Rates can't be removed from [x]. Provide prices for all dimensions in the existing rate card.

Error code	Error message
INVALID_RATE_CARD	Provide dimensions that have the same unit in [x].
INVALID_RATE_CARD	Provide either all metered or all entitled dimensions in [x].
INVALID_RATE_CARD	Provide only entitled dimensions in [x].
INVALID_RATE_CARD	Provide usage based rates for all available metered dimensions in UsageBasedPricingTerm.
INVALID_RATE_CARD	Provide usage based rates for all free trial dimensions.
INVALID_RATE_CARD	Provide prices with up to 3 decimal places in UsageBasedPricingTerm.
INVALID_SELECTOR_DURATION_VALUE	Provide duration between [x] and [y] months.
INVALID_SELECTOR_DURATION_VALUE	Ensure duration granularity is at the day level for metered dimensions.
INVALID_SELECTOR_DURATION_VALUE	Ensure Duration in ConfigurableUpfrontPricingTerm is within the allowed range.
INVALID_SELECTOR_DURATION_VALUE	Provide one or more supported contract durations.
INVALID_SELECTOR_DURATION_VALUE	Provide one or more supported contract durations or a single custom duration.
INVALID_SELECTOR_DURATION_VALUE	Provide Duration in ConfigurableUpfrontPricingTerm that matches the duration between AgreementStartDate and AgreementEndDate.
MISSING_DURATION	Provide Duration in FixedUpfrontPricingTerm.

Error code	Error message
MISSING_MANDATORY_TERMS	FixedUpfrontPricingTerm is only supported when paired with ByolPricingTerm or PaymentScheduleTerm.
MISSING_MANDATORY_TERMS	Provide at least one of [x,y,z].
MISSING_MANDATORY_TERMS	Provide a ByolPricingTerm when using Byol pricing model.
TOO_MANY_GRANTS	Provide up to [x] grants in [y].
TOO_MANY_RATE_CARDS	Only one rate card in ConfigurableUpfrontPricingTerm is allowed for the product.
TOO_MANY_RATE_CARDS	Up to [x] rate cards are allowed in ConfigurableUpfrontPricingTerm for the product.
TOO_MANY_RATES	Provide RateCards within the allowed limits in ConfigurableUpfrontPricingTerm.
TOO_MANY_RATES	Provide RateCards within the allowed limits in UsageBasedPricingTerm.

Update the discoverability of the offer

You can use the Catalog API to control the discoverability of your offer in AWS Marketplace.

You can either choose to set a specific date in the future to restrict the discoverability of your offer or in the past to expire your offer. The UpdateAvailability change type doesn't affect existing agreements.

Note

- You can use the UpdateAvailability change type on a private offer that has already been [published](#) (also known as *released*). If buyers have already accepted the private offer, those existing agreements aren't affected.

- When modifying the `AvailabilityEndDate` of an existing private offer, the [constraints of the agreement duration](#) must be adhered to. If it's not, include an additional `UpdateValidityTerms` change type in this change set to modify the agreement duration to adhere to the new expiration. The `UpdateValidityTerms` change type can be used on a private offer that is either released or not yet released.
- When modifying the `AvailabilityEndDate` of an existing private offer, the [constraints of the payment schedule](#) must be adhered to. If it's not and the private offer is *not yet released*, include an additional `UpdatePaymentScheduleTerms` change type in this change set to modify the payment schedule to adhere to the new expiration. If the private offer is *already released*, you can only make changes to the `AvailabilityEndDate` as long as the new date adheres to the constraints of the payment schedule.

To control the discoverability of your offer, call the `StartChangeSet` API operation with the `UpdateAvailability` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2024-05-31"
      }
    }
  ]
}
```

Provide information for the fields to add the `UpdateAvailability` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **AvailabilityEndDate** (string) (required) – This is the date until when the offer is discoverable and purchasable in AWS Marketplace. You can choose to set a specific date in the future to restrict the availability or in the past to expire the offer. Dates are represented in YYYY-MM-DD format.

A change set is created for your request. The response to this request gives you the ID and ARN for the change set and looks like the following.

Response Syntax

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. It includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the `DescribeChangeSet` action.

Synchronous Validations

The following schema validations are specific to `UpdateAvailability` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
AvailabilityEndDate	Required	422
	Format: "YYYY-MM-DD"	

Asynchronous Errors

The following errors are specific to UpdateAvailability actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more details about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_AVAILABILITY_END_DATE	AvailabilityEndDate isn't supported for public offers.
INVALID_AVAILABILITY_END_DATE	Provide a future AvailabilityEndDate.
INVALID_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate that is before AgreementEndDate.
MISSING_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate that is before the agreement's end date.

Define the expiration date of agreements created using the offer

You can use the Catalog API to define the expiration date details of agreements created using the offer in AWS Marketplace.

This change type doesn't affect existing agreements.

Note

You can use the UpdateValidityTerms change type on a private offer that has already been [published](#) (also known as *released*). If buyers have already accepted the private offer, those existing agreements aren't affected.

For **AMI-based** and **container-based** products, if your private offer [pricing terms](#) include a term type that has a Duration (for example, the term types FixedUpfrontPricingTerm or ConfigurableUpfrontPricingTerm), your AgreementDuration set in this change type must be greater than the following: the number of days from today to the [expiration of the private offer](#) plus the number of days set in the Duration of those term types. This is because after a buyer accepts the private offer and the agreement is created, they can optionally purchase additional entitlements

specified in those term types until the private offer expires. Furthermore, all additional entitlements must end before the agreement does. For example, if the buyer accepts the private offer on the first available day and then purchases entitlements on the last available day, those entitlements must not end after the agreement end date.

To define the expiration date details of agreements created using the offer, call the `StartChangeSet` API operation with the `UpdateValidityTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateValidityTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "ValidityTerm",
            "AgreementDuration": "P12M",
            "AgreementStartDate": "2021-08-01",
            "AgreementEndDate": "2022-08-01"
          }
        ]
      }
    }
  ]
}
```

Provide information for the fields to add the `UpdateValidityTerms` change type:

- **Entity** (object) (required) – Your offer.

- **Type** (string) (required) – The Type is always `Offer@1.0`.
- **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
- **Terms** (array of structures) – List of validity terms that you want to update. Supported validity terms are:
 - **ValidityTerm** (object) – Defines the conditions that will keep an agreement, created from this offer, valid.
 - **Type** (string) – Category of the term being updated. `ValidityTerm`
 - **AgreementDuration** (string) – Defines the duration that the agreement remains active. If `AgreementStartDate` isn't provided, agreement duration is relative to the agreement signature time. The duration is represented in the ISO_8601 format.
 - **AgreementStartDate** (string) – Defines the date when agreement starts. `AgreementStartDate` is represented in YYYY-MM-DD format. The agreement starts at 00:00:00.000 UTC on the date provided. If `AgreementStartDate` isn't provided, agreement start date is determined based on agreement signature time.
 - **AgreementEndDate** (string) – Defines the date when the agreement ends. The `AgreementEndDate` is represented in YYYY-MM-DD format. The agreement ends at 23:59:59.999 UTC on the date provided. If `AgreementEndDate` isn't provided, the agreement end date is determined by the validity of individual terms.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdateValidityTerms actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required	422
Terms[].Type	Required Can only be "ValidityTerm"	422
Terms[].AgreementDuration	Optional Expected format per Selector type: ISO 8601 duration Can be stand alone or paired with AgreementStartDate	422
Terms[].AgreementEndDate	Optional Date must be formatted like "YYYY-MM-DD"	422
Terms[].AgreementStartDate	Optional Date must be formatted like "YYYY-MM-DD" Can only be paired with Agreement EndDate and Agreement Duration	422

Asynchronous Errors

The following errors are specific to `UpdateValidityTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_AGREEMENT	AgreementStartDate can't be in the future when the current agreement to be replaced isn't future dated.
INCOMPATIBLE_AGREEMENT_END_DATE	AgreementEndDate can't be updated after the offer is released.
INCOMPATIBLE_AGREEMENT_START_DATE	AgreementStartDate can't be updated after the offer is released.
INCOMPATIBLE_PRODUCT	AgreementStartDate in the future isn't supported.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the duration between Agreement StartDate and AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementStartDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the duration between Agreement StartDate and AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementDuration matches duration specified in the ResaleAuthorization.
INCOMPATIBLE_TERMS	ValidityTerm isn't supported for public offers.

Error code	Error message
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_AGREEMENT_DURATION	Provide AgreementDuration that is greater than or equal to [x] days.
INVALID_AGREEMENT_END_DATE	Provide a future AgreementEndDate.
INVALID_AGREEMENT_END_DATE	Provide AgreementEndDate that is after or equal to [x].
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is after AvailabilityEndDate.
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is before the AgreementEndDate.
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is within [x] years from today.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with both AgreementDuration and AgreementEndDate isn't supported.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with both AgreementStartDate and AgreementDuration isn't supported in an offer for the product.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with AgreementStartDate isn't supported in an offer for the product.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with only AgreementStartDate isn't supported.
INVALID_AGREEMENT_TIME_INTERVAL	AgreementEndDate isn't supported unless it's used in combination with a future AgreementStartDate or for replacement offers.

Error code	Error message
INVALID_AGREEMENT_TIME_INTERVAL	Provide AgreementStartDate and AgreementEndDate where the difference is less than or equal to [x] years.
MISSING_AGREEMENT_START_DATE	Ensure AgreementStartDate is present in ValidityTerm when used along with ConfigurableUpfrontPricingTerm.

Update payment schedule details

You can use the Catalog API to update payment schedule details for your offer, such as flexible payment schedule, in AWS Marketplace.

Note

You cannot use the UpdatePaymentScheduleTerms change type on an offer that has already been [published](#) (also known as *released*).

The private offer can be accepted any day between the creation of the private offer and its [expiration](#) (set in the AvailabilityEndDate). Only one ChargeDate value of the payment schedule can be a date on or before the last day the buyer can accept the private offer (the private offer expiration date). The remaining values of ChargeDate must be after the private offer expiration, but no later than the end of the agreement if the private offer was accepted immediately. The end of the agreement is based on when the private offer is accepted (creating the agreement) plus the [duration of the agreement](#).

To update payment schedule details for your offer, call the StartChangeSet API operation with the UpdatePaymentScheduleTerms change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
```

```

"ChangeSet": [
  {
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "offer-123456789"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "PaymentScheduleTerm",
          "Schedule": [
            {
              "ChargeDate": "2021-12-01",
              "ChargeAmount": "200.00"
            },
            {
              "ChargeDate": "2022-03-01",
              "ChargeAmount": "250.00"
            }
          ]
        }
      ]
    }
  }
]
}

```

Provide information for the fields to add the `UpdatePaymentScheduleTerms` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Terms** (array of structures) – List of payment terms that you want to update. Supported payment terms are:
 - **PaymentScheduleTerm** (object) – Defines an installment-based pricing model where customers are charged a fixed price on different dates during the agreement validity period.
 - **Type** (string) – Type of the term being updated. This is the object value: `"PaymentScheduleTerm"`.

- **Schedule** (array of structures) – List of the payment schedule where each element defines one installment of payment. It contains the information necessary for calculating the price to be paid and the date on which the customer would be charged.
- **ChargeDate** (string) – The date on which the customer would pay the price defined in this payment schedule term. ChargeDate is represented in YYYY-MM-DD format. Invoices are generated on the date provided.
- **ChargeAmount** (string) – The price that the customer would pay on scheduled date (ChargeDate).

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdatePaymentScheduleTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input Field	Validation Rule	HTTP
Terms	Required	422
	Only <code>PaymentScheduleTerm</code> is allowed	

Input Field	Validation Rule	HTTP
	List size must be less than 2	
Terms[].Type	Required Can only be PaymentScheduleTerm	422
Terms[].PaymentScheduleTerm.CurrencyCode	Required Supported currencies: ["USD", "AUD", "EUR", "GBP", "JPY"]	422
Terms[].PaymentScheduleTerm.Schedule[]	Required	422
Terms[].PaymentScheduleTerm.Schedule[].ChargeAmount	Required Date type is "String" Non-negative decimals with up to 2 decimal places supported	422
Terms[].PaymentScheduleTerm.Schedule[].ChargeDate	Required Date must be formatted like "YYYY-MM-DD"	422

Asynchronous Errors

The following errors are specific to UpdatePaymentScheduleTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
DUPLICATE_CHARGE_DATES	Provide unique charge dates in PaymentScheduleTerm.
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.
INCOMPATIBLE_MARKUP_PERCENTAGE	PaymentScheduleTerm isn't supported when MarkupPercentage is greater than zero (0).
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide term(s) that are compatible with the ResaleAuthorization. Incompatible terms: [PaymentScheduleTerm].
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the total ChargeAmounts in PaymentScheduleTerm is compatible with the ResaleAuthorization.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_CHARGE_DATES	Provide charge dates before AgreementEndDate.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.

Error code	Error message
TOO_MANY_BACKDATED_CHARGES	Provide up to 1 scheduled payment before AvailabilityEndDate.

Modify renewal options

You can use the Catalog API to control renewal options of the agreements that are created using this offer in AWS Marketplace.

For offers created through Catalog API, auto-renewal remains disabled by default until you call the `UpdateRenewalTerms` change type to allow auto-renewal. This change does not affect existing agreements.

To control renewal options of the agreements that are created using this offer, call the `StartChangeSet` API operation with the `UpdateRenewalTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateRenewalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "RenewalTerm"
          }
        ]
      }
    }
  ]
}
```

```
}
```

Provide information for the fields to add the `UpdateRenewalTerms` change type:

- **Entity** (object) (required) – Your offer.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
- **Terms** (array of structures) – List of renewal terms that you want to update. Supported renewal terms are:
 - **RenewalTerm** (object) – Defines that on graceful termination (expiration of the `ValidityTerm`, not buyer or AWS Marketplace cancellation) of the agreement, a new agreement will be created using the accepted terms on the existing agreement. In other words, the agreement will be renewed. Presence of `RenewalTerm` in the offer means that auto-renewal is allowed. Buyers will have the option to accept or decline auto-renewal at the offer acceptance/agreement creation.
 - **Type** (string) – Type of the term being updated. `RenewalTerm`

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateRenewalTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required	422
Terms[].Type	Required Can only be "RenewalTerm"	422

Asynchronous Errors

The following errors are specific to `UpdateRenewalTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more details about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT	RenewalTerm isn't supported in private offers for the product.
INCOMPATIBLE_TERMS	RenewalTerm isn't supported together with PaymentScheduleTerm.
INCOMPATIBLE_TERMS	RenewalTerm isn't supported with the PricingModel.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.

Publish an offer

You can use the Catalog API to merge the information collected from all update change types, and then publish the offer.

Offers remain in a Draft state, until `ReleaseOffer` is called. After the offer is released, it's discoverable in AWS Marketplace.

To publish your offer, call the `StartChangeSet` API operation with the `ReleaseOffer` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {}
    }
  ]
}
```

Provide information for the fields to add the `ReleaseOffer` change type:

- **Entity** (object) – The named type of entity being created. The `Identifier` is your offer ID, and the `Type` is always `Offer@1.0`. For more information, see [Identifier](#).
- **DetailsDocument** (object) – The JSON value of specifics of the request. It must be empty for `ReleaseOffer`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take a few minutes.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `ReleaseOffer` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
DetailsDocument	Must be empty ({})

Asynchronous Errors

The following errors are specific to `ReleaseOffer` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_PRODUCT	First create a public offer for the product.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.

Error code	Error message
INVALID_UPDATE_REQUEST	The requested change can't be performed after the offer is released.
MISSING_AGREEMENT_END_DATE	Provide an AgreementEndDate for replacement offers.
MISSING_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate for private offer.
MISSING_BUYER_ACCOUNTS	Provide PositiveTargeting with BuyersAccounts for offers created using ResaleAuthorization.
MISSING_BUYER_ACCOUNTS	All offers for the product must be private. Provide PositiveTargeting with BuyersAccounts.
MISSING_DESCRIPTION	Set Description before releasing the offer.
MISSING_MANDATORY_TERMS	Add [x] to the offer.
MISSING_MANDATORY_TERMS	Provide a FixedUpfrontPricingTerm when the offer contains a PaymentScheduleTerm.
MISSING_NAME	Set Name before releasing the offer.
TOO_MANY_OFFERS	Only one public free trial offer can be created per product.
TOO_MANY_OFFERS	Only one public offer can be created per product.
INCOMPATIBLE_TARGETING	PreExistingAgreement is only supported for buyer targeted offers.

Describe existing offer details

You can use the Catalog API to describe existing offer details in AWS Marketplace.

To describe existing offer details, call the `DescribeEntity` API operation with the `Offer@1.0` entity type, as shown in the following example.

Request Syntax

```
GET /DescribeEntity?catalog=<Catalog>&entityId=<EntityId> HTTP/1.1
```

Provide information for the fields to add the `DescribeEntity` change type:

- **catalog** (string) – The catalog related to the request. Fixed value: `AWSMarketplace`.
- **entityId** (string) – The unique ID of the offer to describe.

Response Syntax

The response to this request gives you the offer details and looks like the following.

```
{
  "EntityType": "Offer@1.0",
  "EntityIdentifier": "offer-ad8EXAMPLE51@1",
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:111122223333:AWSMarketplace/Offer/offer-ad8EXAMPLE51",
  "LastModifiedDate": "2021-03-10T21:57:16Z",
  "DetailsDocument": {
    "Id": "offer-3rEXAMPLErn",
    "State": "Released",
    "Name": "Test Offer",
    "Description": "Worldwide offer for Test Product",
    "PreExistingAgreement": {
      "AcquisitionChannel": "External",
      "PricingModel": "Contract"
    },
    "ProductId": "prod-ad8EXAMPLE51",
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "If you need to request a refund for software sold by Amazon Web Services, LLC, please contact AWS Customer Service."
      },
    ],
  },
}
```



```
{
  "Type": "LegalTerm",
  "Documents": [
    {
      "Type": "CustomEula",
      "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
    }
  ]
},
{
  "Type": "FreeTrialPricingTerm",
  "Duration": "P30D",
  "Grants": [
    {
      "DimensionKey": "m3.xlarge",
      "MaxQuantity": 10
    },
    {
      "DimensionKey": "m4.xlarge",
      "MaxQuantity": 10
    }
  ]
},
{
  "Type": "ConfigurableUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "Selector": {
        "Type": "Duration",
        "Value": "P365D"
      },
      "RateCard": [
        {
          "DimensionKey": "m3.large",
          "Price": "300.00"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "400.00"
        }
      ]
    },
    {
      "Constraints": {
        "MultipleDimensionSelection": "Allowed",
```

```
        "QuantityConfiguration": "Allowed"
      }
    }
  ],
},
{
  "Type": "UsageBasedPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "RateCard": [
        {
          "DimensionKey": "m3.large",
          "Price": "0.10"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "0.20"
        }
      ]
    }
  ]
},
{
  "Type": "FixedUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "Price": "200.00",
  "Grants": [
    {
      "DimensionKey": "Users",
      "MaxQuantity": 10
    }
  ]
},
{
  "Type": "RecurringPaymentTerm",
  "CurrencyCode": "USD",
  "BillingPeriod": "Monthly",
  "Price": "100.0"
},
{
  "Type": "PaymentScheduleTerm",
  "CurrencyCode": "USD",
  "Schedule": [
```

```
    {
      "ChargeDate": "2020-12-01T00:00:00.000Z",
      "ChargeAmount": "1000.00"
    },
    {
      "ChargeDate": "2021-06-15T00:00:00.000Z",
      "ChargeAmount": "1250.00"
    }
  ]
},
{
  "Type": "ByolPricingTerm"
},
{
  "Type": "RenewalTerm"
}
],
"Rules": [
  {
    "Type": "TargetingRule",
    "PositiveTargeting": {
      "CountryCodes": [
        "US",
        "CA"
      ],
      "BuyerAccounts": [
        "444455556666"
      ]
    },
    "NegativeTargeting": {
      "CountryCodes": [
        "XX"
      ]
    }
  },
  {
    "Type": "AvailabilityRule",
    "AvailabilityEndDate": "2024-08-30T01:56:03.000Z"
  }
]
}
```

The following is information about the fields you see in the `DescribeEntity` response.

- **EntityType** (string) – The named type of the entity, which is `Offer@1.0`.
- **EntityIdentifier** (string) – The identifier of the entity, in the format of `EntityId@RevisionId`.
- **EntityArn** (string) – The ARN associated to the unique identifier for the change set referenced in this request.
- **LastModifiedDate** (string) – The last modified date of the entity, in ISO 8601 format (for example: `2018-02-27T13:45:22Z`).
- **Details** (string) – This stringified JSON object includes the following details of the entity:
 - **Id** (string) – Unique identifier for an offer entity in AWS Marketplace and is generated during the creation of an offer.
 - **State** (string) – The status of the offer.
 - **Name** (string) – The name associated with the offer for better readability to you and your customers. It will be displayed as part of Agreement information as well.
 - **Description** (string) – Description is a free-form text which is meant to be used only by you and will never be exposed to buyers.
 - **PreExistingAgreement** (string) – Determines if this offer is a renewal for an existing agreement with an existing customer for the same underlying product. The existing agreement can be within or outside AWS Marketplace. AWS may audit and verify your offer is a renewal. If AWS is unable to verify your offer, then AWS may revoke the offer and entitlements from your customer.
 - **AcquisitionChannel** (string) – Indicates if the existing agreement was signed outside AWS Marketplace or within AWS Marketplace. Possible values: `External`, `AwsMarketplace`.
 - **PricingModel** (string) – Indicates which pricing model the existing agreement uses. Possible values: `Contract`, `Usage`, `Byol`, `Free`.
 - **ProductId** (string) – The unique identifier of the product being offered.
 - **Terms** (array of structures) – List of terms.
 - **Rules** (array of structures) – List of rules.

Work with resale authorizations using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with Resale Authorizations.

While the *product* describes what is being sold in AWS Marketplace, the *Resale Authorization* (also known as an opportunity) describes the terms and rules regarding how this product is authorized to be resold in AWS Marketplace. The *CPPO* is the target of the Resale Authorization.

A Resale Authorization has a collection of terms and rules to be accepted for a reseller agreement between manufacturers and channel partners. Accepting the terms of the Resale Authorization allows the reseller to create offers for the product per the conditions expressed in the terms.

There are two types of rules in a Resale Authorization:

- **AvailabilityRule** – Controls the lifecycle of the Resale Authorization in AWS Marketplace.
- **PartnerTargetingRule** – Specifies whether the Resale Authorization should be accessible to a specific set of channel partners.

See the following resources:

- For end-to-end labs with working code examples, see [Lab: Authorize a reseller](#) in the *AWS Marketplace seller workshop*.
- For code examples of API requests, see [Python](#) and [Java](#) examples in *AWS Samples* on GitHub.
- For a video on creating resale authorizations, see [Create Resale Authorizations Using the AWS Marketplace Catalog API](#) on YouTube.

The following topics describe how to use the Catalog API to create and update Resale Authorizations:

Topics

- [Resale Authorization prerequisites](#)
- [Create a new Resale Authorization](#)
- [Update buyer targeting](#)
- [Update availability](#)
- [Update the validity of a future dated agreement](#)
- [Update legal resources](#)
- [Update pricing](#)
- [Update payment schedule](#)
- [Update Resale Authorization details](#)

- [Restrict a Resale Authorization](#)
- [Release a Resale Authorization and make it visible to a Channel Partner](#)
- [Describe an existing Resale Authorization](#)

Resale Authorization prerequisites

To use Resale Authorization, both independent software vendors (ISVs) and AWS Marketplace Channel Partners must create a service-linked role that provides resource-sharing permissions to AWS. If both groups don't perform this prerequisite, AWS can't share the authorization resource from the ISV to the AWS Marketplace Channel Partner. For more information, see [Using roles for Resale Authorization for AWS Marketplace](#) in the *AWS Marketplace Seller Guide*.

Create a new Resale Authorization

You can use the Catalog API to create a new Resale Authorization in AWS Marketplace.

If your request is processed successfully, AWS Marketplace Catalog API generates a Resale Authorization in `Draft` state for you. It's an incomplete Resale Authorization and not visible to channel partners in AWS Marketplace.

Use the Update change types to complete the Resale Authorization. After the Resale Authorization is completed, use the `ReleaseResaleAuthorization` change type to complete the Resale Authorization creation process and release the Resale Authorization, which will validate the entire Resale Authorization and make your it visible to channel partners in AWS Marketplace.

To create a Resale Authorization in `Draft` state, call the `StartChangeSet` API operation with the `CreateResaleAuthorization` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "xyz",
      "Entity":
```

```
{
  "Type": "ResaleAuthorization@1.0"
},
"DetailsDocument":
{
  "ProductId": "prod-ad8EXAMPLE51",
  "Name": "Test ResaleAuthorization",
  "Description": "Worldwide ResaleAuthorization for Test Product",
  "ResellerAccountId": "777788889999"
}
]
}
```

Provide information for the input fields to add the `CreateResaleAuthorization` change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
- **DetailsDocument** (object) (required) – Specifics of the request.
 - **ProductId** (string) (required) – Product ID for which to create the resale authorization.
 - **Name** (string) (required) – Name associated with the `ResaleAuthorization` for better readability to you and your channel partners.
 - **Description** (string) (optional) – A free-form text field available to add details about the `ResaleAuthorization`.
 - **ResellerAccountId** (string) (required) – Add targeted channel partner's AWS account who can describe and use this `ResaleAuthorization` to create a private offer.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

When the request is complete (if the Status is SUCCEEDED), a new ResaleAuthorization is generated. Although the SUCCEEDED status indicates that the CreateResaleAuthorization change type call is completed, the ResaleAuthorization status is still in Draft state.

The following shows the response from the [DescribeChangeSet](#) API operation.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef",
  "ChangeSetName": "Submitted by 123456789012",
  "StartTime": "2021-05-27T22:21:26Z",
  "EndTime": "2021-05-27T22:32:19Z",
  "Status": "SUCCEEDED",
  "ChangeSet":
  [
    {
      "ChangeType": "CreateResaleAuthorization",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "ProductId": "prod-ad8EXAMPLE51",
        "Name": "Test ResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "777788889999",
        "BulkRequestId": "84977023-5093-4a66-8b24-ef2c5a2f8b1f"
      },
      "ErrorDetailList":
      []
    }
  ]
}
```

Synchronous Validations

The schema validations are specific to `CreateResaleAuthorization` actions in the AWS Marketplace Catalog API. The validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
ProductId	Required Must not be null or empty Length must be between 1 and 50 characters	422
ProductId	User must be authorized to create ResaleAuthorization for the given product	403
ProductId	Must be an existing product in the catalog and not in Draft state Product should be supported to resell	404
Name	Required Must not be null or empty Length must be between 1 and 100 characters No special characters allowed	422
Description	Optional Length must be between 1 and 255 characters No special characters allowed	422
ResellerAccountId	Required	422

Input field	Validation rule	HTTP code
	Must not be empty	
	AWS account IDs must be in valid format (12-digit number)	
BulkRequestId	Optional	422
	Length must be between 1 and 50 characters	
	Must be in UUID format	
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to `CreateResaleAuthorization` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_RESELLER_ACCOUNT	Provide a valid reseller account.

Update buyer targeting

You can use the Catalog API to update buyers targeting your Resale Authorization in AWS Marketplace.

Any existing targeting options that aren't included in the latest request are removed from the Resale Authorization. This change type is optional for release of the Resale Authorization.

To update buyers targeting your Resale Authorization, call the `StartChangeSet` API operation with the `UpdateBuyerTargetingTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateBuyerTargetingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerTargetingTerm",
            "PositiveTargeting": {
              "BuyerAccounts": [
                "123456789012"
              ]
            }
          }
        ]
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateBuyerTargetingTerms change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always ResaleAuthorization@1.0.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
 - **Terms** (array of structures) (optional) – List of buyers targeting terms that you want to update. If the intentions aren't to target the ResaleAuthorization to any specific buyer, then terms

field can be skipped. By default, `ResaleAuthorization` is targeted to all buyers. Supported terms are:

- **BuyerTargetingTerms** (object) (optional) – Define buyer-specific targeting to your `ResaleAuthorization`.
- **Type** (string) (required) – Category of the term being updated.
- **PositiveTargeting** (object) (required) – Defines the criteria that any buyer's profile should fulfill to be allowed access to the `ResaleAuthorization`.
 - **BuyerAccounts** (array of strings) (optional) – List as optional. You can add the targeted buyer's AWS accounts. If the intention isn't to target `ResaleAuthorization` to specific buyers, then this field should be omitted. By default, all buyers are targeted. Targeted channel partners can choose to create a private offer and target a subset of buyers, if specified.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information with the AWS Marketplace Seller Operations team to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to `UpdateBuyerTargetingTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Terms	Optional Must not be null or empty Only "BuyerTargetingTerm" is allowed in the list List size must be 1 (there is no use case today that requires multiple buyer terms)
BuyerTargetingTerm.PositiveTargeting	Required Must not be empty
BuyerTargetingTerm.PositiveTargeting.BuyerAccounts	Optional AWS account IDs must be in valid format (12-digit number) Must not contain more than 25 accounts
An unknown property	No additional properties are allowed

Asynchronous Errors

The following errors are specific to `UpdateBuyerTargetingTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_BUYER_TARGETING	At least one Buyer account must be present for ResaleAuthorization with PreExistingBuyerAgreement.

Update availability

You can use the Catalog API to limit the availability of how many private offers are created or until what specific time a private offer can be created.

By default, the value is unlimited usage of this Resale Authorization, although you can check the availability under the rule list.

To control the availability and usability of your Resale Authorization, call the StartChangeSet API operation with the UpdateAvailability change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateAvailability",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "AvailabilityEndDate": "2022-05-31",
        "OffersMaxQuantity": 1
      }
    }
  ]
}
```

```
}
```

Provide information for the fields to add the UpdateAvailability change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always ResaleAuthorization@1.0.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
 - **AvailabilityEndDate** (string) (optional) – Define the end date until Channel Partners can leverage the ResaleAuthorization to create an offer. Channel Partners can use this ResaleAuthorization multiple times until the specified end date. Dates are represented in ISO_8601 format.
 - **OffersMaxQuantity** (integer) (optional) – Define the maximum number of private offers that can be created using the ResaleAuthorization. This doesn't define the number of subscriptions.

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information with the AWS Marketplace Seller Operations team to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to UpdateAvailability actions in the AWS Marketplace Catalog API. The validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response

Input field	Validation rule
OffersMaxQuantity	Optional Must be non-negative integer Allowed value only "1" (Currently no use case to support multiple quantity)
AvailabilityEndDate	Optional Must be ISO_8601 formatted Must be date in the future
Availability	Provide either OffersMaxQuantity or AvailabilityEndDate.
An unknown property	No additional properties are allowed

Asynchronous Errors

The following errors are specific to UpdateAvailability actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate that is before all the ChargeDate in ResalePaymentScheduleTerms.
INVALID_AVAILABILITY_END_DATE	Provide a future AvailabilityEndDate.

Update the validity of a future dated agreement

You can use the Catalog API to modify and control a future dated service start date in AWS Marketplace.

This change set is not mandatory to release a Resale Authorization.

To modify and control the product agreement duration of your Resale Authorization, call the `StartChangeSet` API operation with the `UpdateBuyerValidityTerms` change type, as shown in the following example.

Note

Future-dated agreements are only supported for SaaS product types.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateBuyerValidityTerms",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "Terms":
        [
          {
            "Type": "BuyerValidityTerm",
            "MaximumAgreementStartDate": "2024-05-31"
          }
        ]
      }
    }
  ]
}
```

```
]
}
```

Provide information for the input fields to add the `UpdateBuyerValidityTerms` change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
 - **Terms** (array of structures) – List of agreement validity terms that you want to update. Supported terms are:
 - **BuyerValidityTerm** (object) – Defines the availabilities of a service for a product in your ResaleAuthorization.
 - **Type** (string) – Category of term being updated.
 - **MaximumAgreementStartDate** (string) (required) – Define the agreement start date for the product offered. Future dated offers can't exceed this service start date. Dates are represented in ISO_8601 format.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information with the AWS Marketplace Seller Operations team to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to `UpdateBuyerValidityTerms` actions in the AWS Marketplace Catalog API. The validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Terms	Required Must not be null or empty Only "BuyerValidityTerm" is allowed in the list List size must be 1 (there's no use case today that requires multiple service availability terms)
MaximumAgreementStartDate	Required Must not be null or empty Must be future date and shouldn't exceed more than 3 years from now Must be ISO_8601 formatted
An unknown property	No additional properties are allowed

Asynchronous Errors

The following errors are specific to `UpdateBuyerValidityTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PRODUCT	BuyerValidityTerm isn't supported for the product.

Error code	Error message
INVALID_MAXIMUM_AGREEMENT_START_DATE	Provide a future MaximumAgreementStartDate with in allowed limit.

Update legal resources

You can use the Catalog API to replace the existing legal terms completely in AWS Marketplace.

The legal terms that aren't included in the latest request will be removed from the Resale Authorization. `BuyerLegalTerm` contains the EULA which will be included on the final buyer agreement and `LegalTerm` includes the Reseller Contract which will be included in the reseller agreement between the channel partner and the ISV.

To update legal terms of your `ResaleAuthorization`, call the `StartChangeSet` API operation with the `UpdateLegalTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "Terms":
        [
          {
            "Type": "BuyerLegalTerm",
            "Documents":
            [
```

```

        {
            "Type": "CustomEula",
            "Url": "https://my-public-bucket.s3.amazonaws.com/eula-
example12345.txt"
        }
    ],
    {
        "Type": "ResaleLegalTerm",
        "Documents":
        [
            {
                "Type": "CustomResellerContract",
                "Url": "https://my-public-bucket.s3.amazonaws.com/reseller-
example12345.txt"
            }
        ]
    }
]
}

```

Provide information for the fields to add the `UpdateLegalTerms` change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
 - **Terms** (array of structures) (required) – List of legal terms. Supported legal terms are:
 - **BuyerLegalTerm** (object) (required) – Defines the list of text agreements to be proposed to acceptors. For example, the end user license agreement (EULA).
 - **Type** (string) (required) – Category of the term being updated.
 - **Documents** (array of structures) (required) – List of references to legal resources to be proposed to the buyers. For example, the EULA. Each reference is made up of a Type and a URL:
 - **Type** (string) (required) – Type of document. Available document types are:

- **StandardEula** – Standard Contract for AWS Marketplace. For more information, see [SCMP](#) in the *AWS Marketplace Seller Guide*. You don't need to provide a URL for this type because it's managed by AWS Marketplace.
- **EnterpriseEula** – Enterprise Contract for AWS Marketplace. For more information, see DSA in the *AWS Marketplace Seller Guide*. You don't need to provide a URL for this type because it's managed by AWS Marketplace.
- **CustomEula** – Custom EULA provided by you as a manufacturer. A URL for the EULA stored in an accessible S3 bucket is required for this document type.
- **Url** (string) (conditionally required) – A URL to the legal document for buyers to read. This is required when category Type is CustomEula.
- **ResaleLegalTerm** (object) (optional) – Defines the list of text agreements to propose only to channel partners. This term won't be available to buyers.
- **Type** (string) (required) – Category of term being updated.
- **Documents** (array of structures) (required) – List of references to the reseller legal resources to be proposed to the channel partners.
 - **Type** (string) (required) – Category of the document. Available document types are:
 - **StandardResellerContract** – Standard Reseller Contract for AWS Marketplace.
 - **CustomResellerContract** – A custom reseller contract by you as a manufacturer. A URL for the reseller contract is stored in an accessible S3 bucket and is required for this document type.
 - **Url** (string) (conditionally required) – URL to the reseller contract document for channel partners to read. It's required when the Type is CustomResellerContract.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to UpdateLegalTerms actions in the AWS Marketplace Catalog API. The validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required Must not be null or empty	422
Terms[].BuyerLegalTerm	Required Must not be null or empty	422
Terms[].ResaleLegalTerm	Optional Must not be null or empty if present	422
Terms[].BuyerLegalTerm.Documents	Required Must not be null or empty	422
Terms[].BuyerLegalTerm.Documents[].Type	Required Must not be null or empty Allowed values: <ul style="list-style-type: none"> StandardEula EnterpriseEula CustomEula 	422

Input field	Validation rule	HTTP code
Terms[].BuyerLegalTerm.Documents[].Url	Required and must be a valid URL when "Type" is "CustomEula" Must not be provided when "Type" is one of ["StandardEula", "EnterpriseEula"]	422
Terms[].ResaleLegalTerm.Documents	Required Must not be null or empty	422
Terms[].ResaleLegalTerm.Documents[].Type	Required Must not be null or empty Allowed values: <ul style="list-style-type: none"> • StandardEula • CustomResellerContract 	422
Terms[].ResaleLegalTerm.Documents[].Url	Required and must be a valid URL when "Type" is "CustomResellerContract" Must not be provided when "Type" is one of ["StandardContract"]	422
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to UpdateLegalTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet, after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_BUYER_LEGAL_DOCUMENTS	Provide URLs for buyer legal documents stored in accessible S3 buckets.
INVALID_RESALE_LEGAL_DOCUMENTS	Provide URLs for resale legal documents stored in accessible S3 buckets.
MISSING_MANDATORY_TERMS	Provide a BuyerLegalTerm.

Update pricing

You can use the Catalog API to replace the existing pricing terms completely in AWS Marketplace.

Pricing terms that aren't included in the latest request will be removed from the Resale Authorization. You can update the discounted pricing for your product through this API.

To update pricing details for your Resale Authorizations, call the `StartChangeSet` API operation with the `UpdatePricingTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "PricingModel": "Contract",
        "Terms":
        [
```

```
{
  "Type": "ResaleUsageBasedPricingTerm",
  "CurrencyCode": "USD",
  "RateCards":
  [
    {
      "RateCard":
      [
        {
          "DimensionKey": "m3.large",
          "Price": "0.10"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "0.20"
        }
      ]
    }
  ],
},
{
  "Type": "ResaleConfigurableUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "RateCards":
  [
    {
      "Selector":
      {
        "Type": "Duration",
        "Value": "P12M"
      },
      "RateCard":
      [
        {
          "DimensionKey": "m3.large",
          "Price": "300"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "400"
        }
      ]
    },
    "Constraints":
    {
```

```
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    }
}
],
},
{
    "Type": "ResaleFixedUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "Duration": "P2M",
    "Price": "200.0",
    "Grants":
    [
        {
            "DimensionKey": "Users",
            "MaxQuantity": 10
        }
    ]
}
]
}
}
]
```

Provide information for the fields to add the `UpdatePricingTerms` change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
- **PricingModel** (string) (required) – Pricing model for your offer. Possible values for pricing model are:
 - **Usage** – Usage-based pricing model where buyers will be billed for their usage of your product.
 - **Contract** – In the contract-based pricing model, buyers are either billed in advance for the use of your product or offered a flexible payment schedule. Buyers can also pay for additional usage above their contract. Channel partners can add their markup to this payment schedule and pricing for each dimension.

- **Terms** (array of structures) (required) – List of pricing terms that you want to update. Supported pricing terms are:
 - **ResaleUsageBasedPricingTerm** (object) – Defines a pay-as-you-go (PAYG) pricing model where the customers are charged based on product usage.
 - **Type** (string) (required) – Category of the term.
 - **CurrencyCode** (string) – Defines the currency for prices mentioned in this term. Currently, only USD is supported.
 - **RateCards** (array of structures) – List of rate cards.
 - **RateCard** (array of structures) – A rate card defines the per-unit rates for the product dimensions.
 - **DimensionKey** (string) – Dimension for which the given entitlement applies. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
 - **Price** (string) – Per unit price for the product dimension which is used for calculating the amount to be charged.
 - **Constraints** (object) (optional) – Defines limits on how the term can be configured by acceptors.
 - **MultipleDimensionSelection** (string) (optional) – Determines if buyers are allowed to select multiple dimensions in the rate card. Possible values are Allowed and Disallowed. Default value is Allowed.
 - **QuantityConfiguration** (string) (optional) – Determines if acceptors are allowed to configure quantity for each dimension in rate card. Possible values are Allowed and Disallowed. Default value is Allowed.
 - **ResaleFixedUpfrontPricingTerm** (object) – Defines a pre-paid pricing model where the customers are charged a fixed upfront amount.
 - **Type** (string) (required) – Category of the term being updated.
 - **CurrencyCode** (string) – Defines the currency for prices mentioned in this term. Defines the currency for the prices mentioned in this term. USD, AUD, EUR, GBP, and JPY are supported.
 - **Price** (string) (required) – Fixed amount to be charged to the customer when this term is accepted.
 - **Duration** (string) (required) – Contract duration of the ResaleAuthorization. This field supports the ISO 8601 format.

- **Grants** (array of structures) (required) – Entitlements that will be granted to the acceptor of fixed upfront pricing as part of agreement execution.
- **DimensionKey** (string) (required) – Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
- **MaxQuantity** (integer) (required) – Maximum amount of capacity that the buyer can be entitled to the given dimension of the product. If MaxQuantity is not provided, the buyer will be able to use an unlimited amount of the given dimension.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdatePricingTerms` actions in the AWS Marketplace Catalog API. The validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Terms	Required
	Must not be null or empty

Input field	Validation rule
	<p>Each term must present only single time</p> <p>Allowed terms:</p> <ul style="list-style-type: none"> • * ResaleUsageBasedPricingTerm • * ResaleConfigurableUpfrontPricingTerm • * ResaleFixedupfrontPricingTerm
Terms[].ResaleUsageBasedPricingTerm.CurrencyCode	<p>Required</p> <p>Allowed values: USD</p>
Terms[].ResaleUsageBasedPricingTerm.Validity	<p>Required</p> <p>Must not be null or empty</p> <p>Expected format: ISO 8601 duration</p>
Terms[].ResaleUsageBasedPricingTerm.RateCards	<p>Required</p> <p>Must not be null or empty</p>
Terms[].ResaleUsageBasedPricingTerm.RateCards[].DimensionKey	<p>Required</p> <p>Must not be null or empty</p> <p>Length must be between 1 and 60</p>
Terms[].ResaleUsageBasedPricingTerm.RateCards[].Price	<p>Required</p> <p>Must not be null or empty</p> <p>Data type is "String"</p> <p>Must be non-negative</p> <p>Support up to 6 Decimal</p> <p>No special characters supported</p>

Input field	Validation rule
Terms[].ResaleConfigurableUpfrontPricingTerm.CurrencyCode	Required Allowed values: ["USD", "AUD", "EUR", "GBP", "JPN"]
Terms[].ResaleConfigurableUpfrontPricingTerm.RateCards[].Selector.Type	Required Must not be null or empty Allowed values: Duration
Terms[].ResaleConfigurableUpfrontPricingTerm.RateCards[].Selector.Value	Required Must not be null or empty Expected format: ISO 8601 duration
Terms[].ResaleConfigurableUpfrontPricingTerm.RateCards[].RateCard.DimensionKey	Required Must not be null or empty Length must be between 1 and 60
Terms[].ResaleConfigurableUpfrontPricingTerm.RateCards[].RateCard.Price	Required Must not be null or empty Data type is "String" Must be non-negative Support up to 6 Decimal No special characters supported
Terms[].ResaleConfigurableUpfrontPricingTerm.RateCards[].Constraints	Optional

Input field	Validation rule
Terms[].ResaleFixedUpfrontPricingTerm.CurrencyCode	Required Allowed values: ["USD", "AUD", "EUR", "GBP", "JPN"]
Terms[].ResaleFixedUpfrontPricingTerm.Price	Required Must not be null or empty Data type is "String" Must be non-negative Support up to 6 Decimal No special characters supported Allowed values: 0.0
Terms[].ResaleFixedUpfrontPricingTerm.Duration	Required Must not be null or empty Expected format: ISO 8601 duration
Terms[].ResaleFixedUpfrontPricingTerm.Grants[].DimensionKey	Required Must not be null or empty Length must be between 1 and 60
Terms[].ResaleFixedUpfrontPricingTerm.Grants[].MaxQuantity	Required Must not be null or empty
An unknown property	No additional properties are allowed

Asynchronous Errors

The following errors are specific to `UpdatePricingTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_CURRENCY_CODE	Provide the same <code>CurrencyCode</code> across all pricing and payment terms.
INCOMPATIBLE_PRODUCT	Use existing, available dimensions in the product in [x].
DUPLICATE_DIMENSION_KEYS	Provide rate card with a unique list of dimension keys in [x]
INVALID_RATE_CARD	Provide dimensions that have the same unit in [x]
INVALID_RATE_CARD	Provide a rate card for only metered dimensions in <code>ResaleUsageBasedPricingTerm</code> .
INVALID_RATE_CARD	Provide usage based rates for all available metered dimensions in <code>ResaleUsageBasedPricingTerm</code> .
TOO_MANY_RATES	Provide <code>RateCards</code> within the allowed limits in <code>ResaleUsageBasedPricingTerm</code> .
DUPLICATE_SELECTORS	Provide a unique list of <code>Selectors</code> in <code>ResaleConfigurableUpfrontPricingTerm</code> .
INVALID_RATE_CARD	<code>ConfigurableUpfrontPricingTerm</code> is missing one or more dimension keys for duration [x]. Provide prices for the same set of dimension keys for all durations.
INVALID_RATE_CARD	Provide either all metered or all entitled dimensions in [x].

Error code	Error message
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	Set MultipleDimensionSelection and QuantityConfiguration to Disallowed in ResaleConfigurableUpfrontPricingTerm for the PricingModel.
TOO_MANY_RATE_CARDS	Only one rate card in ConfigurableUpfrontPricingTerm is allowed for the product.
INCOMPATIBLE_TERMS	The following terms aren't compatible with the PricingModel: [x,y,z].
TOO_MANY_RATES	Provide RateCards within the allowed limits in [x term].
TOO_MANY_GRANTS	Provide up to [N] grants in [x term].
INVALID_SELECTOR_DURATION_VALUE	Provide duration between [x] and [y] months in ResaleConfigurableUpfront
TOO_MANY_GRANTS	Provide duration between [x] and [y] months.
INVALID_SELECTOR_DURATION_VALUE	Ensure duration granularity is at the day level for metered dimensions in ResaleConfigurableUpfront
INVALID_DURATION	Ensure duration granularity is at the day level for metered dimensions in FixedUpfront.
INVALID_RATE_CARD	Provide only entitled dimensions in [x].
MISSING_DURATION	Provide a Duration in [x].
DUPLICATE_DIMENSION_KEYS	Provide Grants with a unique list of dimension keys in [x].
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.

Error code	Error message
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.

Update payment schedule

You can use the Catalog API to change payment-associated details, such as a flexible payment schedule, in AWS Marketplace.

To update payment-associated details for your Resale Authorization, call the `StartChangeSet` API operation with the `UpdatePaymentScheduleTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdatePaymentScheduleTerms",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "Terms":
```

```
[
  {
    "Type": "ResalePaymentScheduleTerm",
    "CurrencyCode": "USD",
    "Schedule":
      [
        {
          "ChargeDate": "2021-12-01",
          "ChargeAmount": "200.00"
        },
        {
          "ChargeDate": "2022-03-01",
          "ChargeAmount": "250.00"
        }
      ]
  }
]
```

Provide information for the fields to add the UpdatePaymentScheduleTerms change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always ResaleAuthorization@1.0.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request.
- **Terms** (array of structures) – List of payment terms that you want to update. Supported payment terms are:
 - **ResalePaymentScheduleTerm** (object) – Defines an installment-based pricing model where the customers are charged a fixed price on different dates during the agreement validity period.
 - **Type** (string) – Category of the term being updated.
 - **CurrencyCode** (string) (required) – Defines the currency for the payment mentioned in the schedule. USD, AUD, EUR, GBP, and JPY are supported.

- **Schedule** (array of structures) – List of the payment schedule where each element defines one installment of payment. It contains the information necessary for calculating the price to be paid and the date on which the customer would be charged.
- **ChargeDate** (string) (required) – The date the customer would pay the price defined in this payment schedule term. This field supports the ISO 8601 format.
- **ChargeAmount** (string) (required) – The price the customer would pay on a scheduled date (ChargeDate).

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to `UpdatePaymentScheduleTerms` actions in the AWS Marketplace Catalog API. The validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP
Terms.Type	Required	422
	Not supported for [x] product	

Input field	Validation rule	HTTP
	Allowed terms: ResalePaymentScheduleTerm	
Terms[].CurrencyCode	Required Allowed values: USD	422
Terms[].ResalePaymentScheduleTerm.Schedule	Required Length must be between 1 and 60	422
Terms[].ResalePaymentScheduleTerm.Shedule.ChargeDate	Required Must be in ISO 8601 format Date must be in the future	422
Terms[].ResalePaymentScheduleTerm.Shedule.ChargeAmount	Required Must be non-negative	422
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to UpdatePaymentScheduleTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing, or more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_TERMS	OffersMaxQuantity and AvailabilityEndDate must be present with ResalePaymentScheduleTerm.
TOO_MANY_SCHEDULED_PAYMENTS	Provide up to 60 scheduled payments in ResalePaymentScheduleTerm.

Error code	Error message
DUPLICATE_CHARGE_DATES	Provide unique charge dates in ResalePaymentScheduleTerm.
INVALID_CHARGE_DATES	Provide a future ChargeDate.
INVALID_CHARGE_DATES	Provide a last charge date that is before [x].
MISSING_MANDATORY_TERMS	Provide a ResaleFixedUpfrontPricingTerm and ResalePaymentScheduleTerm together.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.

Update Resale Authorization details

You can use the Catalog API to update Resale Authorization details in AWS Marketplace.

To update Resale Authorization details, call the `StartChangeSet` API operation with the `UpdateInformation` change type, as shown in the following example.

Note

The UpdateInformation change type only updates the sections provided in the request; all other information remains unchanged.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateInformation",
      "Entity":
      {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument":
      {
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "PreExistingBuyerAgreement":
        {
          "AcquisitionChannel": "AwsMarketplace",
          "PricingModel": "Contract"
        }
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateInformation change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always ResaleAuthorization@1.0.

- **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Details of the request, including the information you want to update information for the Resale Authorization.
- **Name** (string) (optional) – The name associated with the ResaleAuthorization for better readability to you and your channel partners.
- **Description** (string) (optional) – The description is free-form text where you can add details about the ResaleAuthorization.
- **PreExistingBuyerAgreement** (object) (optional) – Determines if this offer is a renewal for an existing agreement with an existing customer for the same underlying product. The existing agreement can be within or outside AWS Marketplace. AWS may audit and verify your offer is a renewal. If AWS is unable to verify your offer, then AWS may revoke the offer and entitlements from your customer.
 - **AcquisitionChannel** (string) (required) – Indicates if the existing buyer agreement was signed outside AWS Marketplace or in AWS Marketplace.

Possible values: External, AwsMarketplace

- **PricingModel** (string) (required) – Indicates which pricing model the exiting agreement uses.

Possible values: Contract, Usage, BYOL, Free

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet, and the request will fail with an HTTP error if the input does not meet the following requirements.

Input field	Validation rule	HTTP code
Name	Optional Must not be null or empty Length must be between 1 and 100 characters Pattern <code>^[A-Za-z0-9]*\$</code> No special character or white space allowed	422
Description	Optional Length must be between 1 and 255 characters Pattern <code>^[A-Za-z0-9\\s]*\$</code> No special characters allowed	422
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to UpdateInformation actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. or more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_BUYER_TARGETING	At least one Buyer account must be present for ResaleAuthorization with PreExistingBuyerAgreement.

Restrict a Resale Authorization

You can use the Catalog API to set restrict rules to a Resale Authorization in AWS Marketplace.

A restricted Resale Authorization can no longer be used by a channel partner to create a private offer. An existing private offer won't be impacted.

To restrict your Resale Authorization, call the StartChangeSet API operation with the RestrictResaleAuthorization change type, as shown in the following example.

Important

This is a non-reversible operation. After the Resale Authorization is marked as Restricted, it can't be in an Active state again.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-123456789"
      },
      "DetailsDocument": {}
    }
  ]
}
```

Provide information for the fields to add the `RestrictResaleAuthorization` change type:

- **Entity** (object) (required) – Your Resale Authorization.
 - **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
 - **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request. It must be an empty object for `RestrictResaleAuthorization`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to `RestrictResaleAuthorization` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
<code>DetailsDocument</code>	Must be empty	422
<code>RestrictResaleAuthorization</code>	Expired <code>ResaleAuthorization</code> can't be marked as <code>Restricted</code>	422

Input field	Validation rule	HTTP code
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to `RestrictResaleAuthorization` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing, or more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_STATUS	Expired <code>ResaleAuthorization</code> can't be marked as restricted.

Release a Resale Authorization and make it visible to a Channel Partner

You can use the Catalog API to initiate your `ResaleAuthorization` to an `Active` state.

`ReleaseResaleAuthorization` makes your `Resale Authorization` active so that a Channel Partner can use your `Resale Authorization` to create private offers.

To release your `Resale Authorization`, call the `StartChangeSet` API operation with the `ReleaseResaleAuthorization` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity":
      {
```

```
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "resaleauthz-123456789"
  },
  "DetailsDocument": {}
}
]
```

Provide information for the fields to add the `ReleaseResaleAuthorization` change type:

- **Entity** (object) (required) – Your Resale Authorization.
- **Type** (string) (required) – The Type is always `ResaleAuthorization@1.0`.
- **Identifier** (string) (required) – Your Resale Authorization ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – Specifics of the request. It must be empty for `ReleaseResaleAuthorization`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The schema validations are specific to `ReleaseResaleAuthorization` actions in the AWS Marketplace Catalog API. The validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
An unknown property	No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to `ReleaseResaleAuthorization` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more details about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
MISSING_MANDATORY_TERMS	Provide a <code>BuyerLegalTerm</code> .
MISSING_MANDATORY_TERMS	Provide a <code>PricingTerm</code> .
INCOMPATIBLE_PRODUCT	Use an active product in limited or public state.
INCOMPATIBLE_PRICING_TERM	<code>PaymentScheduleTerm</code> and <code>FixedUpfrontPricingTerm</code> must be present together.
INCOMPATIBLE_BUYER_TARGETING	At least one Buyer account must be present for <code>ResaleAuthorization</code> with <code>PreExistingBuyerAgreement</code> .
MISSING_MANDATORY_TERMS	Provide at least one of [x,y,z].
INCOMPATIBLE_STATUS	[x] request can't be performed after the resale authorization is released.

Describe an existing Resale Authorization

To describe Resale Authorization details, call the `DescribeEntity` API operation with the `ResaleAuthorization@1.0` entity type, as shown in the following example.

Request Syntax

```
GET /DescribeEntity?catalog=<Catalog>&entityId=<EntityId> HTTP/1.1
```

Provide information for the fields to add the DescribeEntity change type:

- **catalog** (string) – The catalog related to the request. Fixed value: AWSMarketplace.
- **entityId** (string) – The unique ID of the ResaleAuthorization to describe.

Response Syntax

The response to this request gives you the offer details and looks like the following.

```
{
  "EntityType": "ResaleAuthorization@1.0",
  "EntityIdentifier": "resaleauthz-123456789",
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:111122223333:AWSMarketplace/ResaleAuthorization/resaleauthz-123456789",
  "LastModifiedDate": "2021-03-10T21:57:16Z",
  "DetailsDocument": {
    "Name": "TestResaleAuthorization",
    "Description": "ResaleAuthorization for Test Product",
    "ProductId": "prod-ad8EXAMPLE51",
    "ProductName": "TestProduct",
    "Status": "Active", /*Draft, Active, Restricted*/
    "PreExistingBuyerAgreement": {
      "AcquisitionChannel": "Unknown",
      "PricingModel": "Unknown"
    },
    "CreatedDate": "2023-07-18T16:39:31.335Z",
    "ManufacturerLegalName": "ChannelCAPI.Inc",
    "ManufacturerAccountId": "123456789012",
    "Dimensions": [
      {
        "Name": "Protected Resources",
        "Description": "Additional 100 protected resources",
        "Key": "hundredresources",
        "Unit": "Units",
        "Types": [
          "Entitled"
        ]
      }
    ],
    "OfferDetails": {
```



```

    "OfferExtendedStatus": "Not Started", /* Not Started, Completed-Used, Completed-
Usable*/
    "OfferCreatedCount": 0
  },
  "Terms": [
    {
      "Type": "ResaleUsageBasedPricingTerm",
      "Id": "term_id_placeholder",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "RateCard": [
            {
              "DimensionKey": "resource_number",
              "Price": "0.05"
            },
            {
              "DimensionKey": "scanned_data",
              "Price": "0.05"
            }
          ]
        }
      ]
    },
    {
      "Type": "ResaleConfigurableUpfrontPricingTerm",
      "Id": "term_id_placeholder",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P24M"
          },
          "RateCard": [
            {
              "DimensionKey": "hundredresources",
              "Price": "0.04"
            },
            {
              "DimensionKey": "tenTBData",
              "Price": "0.03"
            }
          ]
        }
      ]
    }
  ]
}

```

```
        "DimensionKey": "channel_custom",
        "Price": "0.02"
    }
],
"Constraints": {
    "MultipleDimensionSelection": "Allowed",
    "QuantityConfiguration": "Allowed"
}
}
]
},
{
    "Type": "ResaleFixedUpfrontPricingTerm",
    "Id": "term-sdh27fb2",
    "CurrencyCode": "USD",
    "Duration": "P180D",
    "Price": "0.0",
    "Grants": [
        {
            "DimensionKey": "sdf73rbns93nl120d10xm1",
            "MaxQuantity": 1
        }
    ]
},
{
    "Type": "ResalePaymentScheduleTerm",
    "Id": "term-sdh27fb2",
    "CurrencyCode": "USD",
    "Schedule": [
        {
            "ChargeDate": "2018-07-01T00:00:00.000Z",
            "ChargeAmount": "200.00"
        },
        {
            "ChargeDate": "2019-05-01T00:00:00.000Z",
            "ChargeAmount": "200.00"
        }
    ]
},
{
    "Type": "BuyerLegalTerm",
    "Id": "term_id_placeholder",
    "Documents": [
        {
```

```

        "Type": "StandardEula",
        "Url": "https://resale-auth-legal-terms-iad-beta.s3.us-
east-1.amazonaws.com/09ae57d6-c75a-3a4c-aadf-9b866bae64ab/a85cace8-6d9d-40ca-
a053-78fc265479bf?isSigned=yes"
    }
]
},
{
    "Type": "ResaleLegalTerm",
    "Id": "term_id_placeholder",
    "Documents": [
        {
            "Type": "StandardResellerContract",
            "Url": "https://resale-auth-legal-terms-iad-beta.s3.us-
east-1.amazonaws.com/09ae57d6-c75a-3a4c-aadf-9b866bae64ab/bed55b56-7ab4-4c4c-
b633-3bf4f6efcb98?isSigned=yes"
        }
    ]
},
{
    "Type": "BuyerValidityTerm",
    "Id": "term_id_placeholder",
    "MaximumAgreementStartDate": "2023-09-25T23:59:59.000Z"
},
{
    "Type": "BuyerTargetingTerm",
    "Id": "term_id_placeholder",
    "PositiveTargeting": {
        "BuyerAccounts": [
            {
                "AwsAccountId": "444455556666"
            }
        ]
    }
}
],
"Rules": [
    {
        "Type": "AvailabilityRule",
        "Id": "availability_rule_id_placeholder",
        /* If the AvailabilityEndDate and OffersMaxQuantity not present Usage will be
Unlimited*/
        "Usage": "Limited",
        "AvailabilityEndDate": "2022-05-31T23:59:59Z",
    }
]
}

```

```
    "OffersMaxQuantity": 1
  },
  {
    "Type": "PartnerTargetingRule",
    "Id": "partner_targeting_rule_id_placeholder",
    "ResellerAccountId": "777777777777",
    "ResellerLegalName": "ChannelCAPICP.Inc"
  }
]
}
}
```

The following is information about the fields you see in the `DescribeEntity` response.

- **EntityType** (string) – The named type of the entity, which is `ResaleAuthorization@1.0`.
- **EntityIdentifier** (string) – The identifier of the entity, in the format of `EntityId@RevisionId`.
- **EntityArn** (string) – The ARN associated to the unique identifier for the change set referenced in this request.
- **LastModifiedDate** (string) – The last modified date of the entity, in ISO 8601 format (`2018-02-27T13:45:22Z`).
- **DetailsDocument** (object) (required) – This JSON string includes the details of the entity.
 - **Name** (string) – Name associated with the `ResaleAuthorization` for better readability to you and your Channel Partners. It's displayed as part of the Agreement information.
 - **Description** (string) – Description is a free-form text which is meant to be used only by you and will never be exposed to buyers.
 - **ProductId** (string) – Description is a free-form text which is meant to be used only by you and will never be exposed to buyers.
 - **AgreementToken** (string) – Generated from content in `ResaleAuthorization`. It contains information about terms, rules, and proposer while creating an agreement. It's used for authorization checks and validations during procurement.
 - **Terms** (array of structures) – List of terms presented for acceptance.
 - **Rules** (array of structures) – List of rules or set of instructions.

Work with channel partner private offers using the AWS Marketplace APIs

You can use the AWS Marketplace Catalog API to automate tasks for working with channel partner private offers (CPPOs).

When you create or update a CPPO, the draft offer will contain the terms and rules from a Resale Authorization and will be invisible to the buyer. It's possible, but not required, to involve multiple personas in your organization to create a private offer.

For example, one persona can be responsible for updating prices while a second persona can be responsible for updating the payment schedule. Then, a third persona can be responsible for updating legal terms. You can give a persona permission to update certain parts of an offer. However, you can give only read permissions to Resale Authorizations.

As a prerequisite for calling change types, you must have received one or more Resale Authorizations and be familiar working with AWS Marketplace Catalog API.

For more information, see [Channel partner private offers](#) in the *AWS Marketplace Seller Guide*.

The following topics describe how to use the Catalog API to create and update CPPOs:

Topics

- [CPPO prerequisites](#)
- [Create a CPPO](#)
- [Create a channel partner private replacement offer](#)
- [Update markup](#)
- [Update targeting configuration](#)
- [Update legal resources](#)
- [Update the discoverability of the CPPO](#)
- [Define the expiration date of agreements](#)
- [Update pricing](#)
- [Update payment schedule details](#)
- [Publish the CPPO](#)
- [Define an existing CPPO](#)

CPPO prerequisites

Service-linked role for ResaleAuthorization (SLR) setup is a mandatory pre-requisite to use resale authorization to create a CPPO. To use Resale Authorization, both independent software vendors (ISVs) and AWS Marketplace Channel Partners must create a service-linked role that provides resource-sharing permissions to AWS. If both groups don't perform this prerequisite, AWS can't share the authorization resource from the ISV to the AWS Marketplace Channel Partner. For more information, see [Using roles for Resale Authorization for AWS Marketplace](#) in the *AWS Marketplace Seller Guide*.

Create a CPPO

You use a Resale Authorization targeted to you to create a channel partner private offer (CPPO) in Draft state in AWS Marketplace.

If your request is processed successfully, AWS Marketplace Catalog API generates an offer in Draft state for you with Resale Authorization terms. You can use `DescribeEntity` to see the terms applied to the draft offer from Resale Authorization. This is an incomplete offer and not visible to buyers in AWS Marketplace. You then use change types associated with the CPPO to complete the offer.

After the offer is completed, you use the `ReleaseOffer` change type to complete the offer creation process and release the offer. This will validate the entire offer and make your offer visible to buyers in AWS Marketplace.

To create a channel partner private offer, call the `StartChangeSet` API operation with the `CreateOfferUsingResaleAuthorization` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
```

```
    },
    "DetailsDocument": {
      "ResaleAuthorizationId": "resaleauthz-123456789",
      "Name": "Test Offer"
    }
  }
]
```

Provide information for the fields to add the `CreateOfferUsingResaleAuthorization` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **ResaleAuthorizationId** (string) (required) – The unique identifier that includes product, terms, and rules that are being offered. Channel partners can add additional terms and rules using update change types. ResaleAuthorization must be available and targeted to you as a partner.
 - **Name** (string) (optional) – The name associated with the offer for better readability. It is displayed as a part of the agreement information.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

The response to this request gives you the status of the request. If the status is `SUCCEEDED`, then a new `OfferId` is generated.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef",
  "ChangeSetName": "Submitted by 123456789012",
  "StartTime": "2021-05-27T22:21:26Z",
  "EndTime": "2021-05-27T22:32:19Z",
  "Status": "SUCCEEDED",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-123456789",
        "Name": "Test Offer"
      },
      "ErrorDetailList": []
    }
  ]
}
```

You can use the GET DescribeEntity request to describe the draft offer rules and terms created from ResaleAuthorization in the AWS Marketplace Catalog API Reference. For more information, see [DescribeChangeSet](#).

Terms and rules from ResaleAuthorization

- **LegalTerms** – Provisions describing legal terms, such as the EULA in the ResaleAuthorization will be added to the draft offer. You can add legal terms using the UpdateLegalTerms change type. For more information, see [UpdateLegalTerms](#).
- **PricingTerms** – All the pricing terms (ConfigurableUpfrontPricingTerm, FixedUpfrontPricingTerm, UsageBasedPricingTerm, PaymentScheduleTerms) described by the Manufacturer in the ResaleAuthorization will be added to the draft offer. You can choose to increase the pricing (for each dimension) for your targeted buyers using the UpdateMarkup change type. For more information, see [UpdateMarkup](#) in this guide.
- **PaymentTerms** – If the manufacturer has defined the Future Payment Schedule in the ResaleAuthorization, then you will be able to see the payment terms in the draft offer.

You can choose to increase the payment schedule amount for your targeted buyers using the UpdateMarkup change type. If you want to set the payment schedule for your buyers, you can use UpdatePaymentScheduleTerms. For more information, see [UpdatePaymentScheduleTerms](#).

- **TargetingRule** – If the ResaleAuthorization is targeted to specific buyers, then channel partners can give private offers to a subset of buyers using PositiveTargeting. By default, the rule will include all the buyers from ResaleAuthorization. You can select specific buyers and update the draft offer using the UpdateTargeting change type. For more information, see [UpdateTargeting](#).

```
{
  "EntityType": "Offer@1.0",
  "EntityIdentifier": "offer-a5EXAMPLEEwzpu@1",
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:444555666777:AWSMarketplace/Offer/offer-a5oEXAMPLEEzpu",
  "LastModifiedDate": "2021-03-10T21:57:16Z",
  "DetailsDocument": {
    "Id": "offer-3rb23tu92rn",
    "Name": "Test Offer",
    "Description": "Worldwide private offer for Test Product",
    "ProductId": "prod-0bc848d78b51",
    "ResaleAuthorizationId": "resaleauthz-123456789",
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
          }
        ]
      }
    ],
  },
  {
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P12M"
        }
      }
    ],
  },
}
```

```
    "RateCard": [
      {
        "DimensionKey": "m3.large",
        "Price": "300.00"
      },
      {
        "DimensionKey": "m4.xlarge",
        "Price": "400.00"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
],
{
  "Type": "UsageBasedPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "RateCard": [
        {
          "DimensionKey": "m3.large",
          "Price": "0.10"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "0.20"
        }
      ]
    }
  ]
},
{
  "Type": "PaymentScheduleTerm",
  "CurrencyCode": "USD",
  "Schedule": [
    {
      "ChargeDate": "2020-12-01T00:00:00.000Z",
      "ChargeAmount": "1000.00"
    },
    {
```

```

        "ChargeDate": "2021-06-15T00:00:00.000Z",
        "ChargeAmount": "1250.00"
    }
  ]
}
],
"Rules": [
  {
    "Type": "TargetingRule",
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111222333444"
      ]
    }
  }
]
}
}
}

```

Synchronous Validations

The following schema validations are specific to `CreateOfferUsingResaleAuthorization` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Name	Required Length must be between 1 and 150 characters	422
Description	Required Length must be between 1 and 255 characters	422
ResaleAuthorizationId	Required	422

Input field	Validation rule	HTTP code
	Length must be between 1 and 50 characters	
ResaleAuthorizationId	ResaleAuthorization must be targeted to the channel partner.	422
ResaleAuthorizationId	ResaleAuthorization must be active	422
Channel Partner	Channel Partner must be paid seller in AWS Marketplace	422

Asynchronous Errors

The following errors are specific to `CreateOfferUsingResaleAuthorization` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_RESALE_AUTHORIZATION	Use a ResaleAuthorization in active state.

Create a channel partner private replacement offer

You can use the Catalog API to create a channel partner private replacement offer in AWS Marketplace.

You use the `ResaleAuthorization` targeted to you and an Agreement of which you are the proposer to create a channel partner private replacement offer in Draft state in the Catalog API by calling `StartChangeSet` with the `CreateReplacementOfferUsingResaleAuthorization` change type, as shown in the following example. Replacement offers can be used to replace an agreement from a previous offer before it ends.

`CreateReplacementOfferUsingResaleAuthorization` will create a draft offer with the agreement acceptor in targeting. This targeting cannot be changed afterwards. The draft offer will also contain the source offer id of the agreement.

To create a channel partner private replacement offer, call the `StartChangeSet` API operation with the `CreateReplacementOfferUsingResaleAuthorization` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateReplacementOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ResaleAuthorizationId": "2bd2c761-3b7f-3771-a9a7-e8ad36517698",
        "Name": "CAPI-saas-abo-contract-fps",
        "AgreementId": "agmt-f2ooEXAMLEamt7mjj0j59gu"
      }
    }
  ]
}
```

Provide information for the fields to add the `CreateReplacementOfferUsingResaleAuthorization` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **ResaleAuthorizationId** (string) (required) – `ResaleAuthorizationId` is the unique identifier which includes product, terms and rules are being offered. Channel partners can add additional terms and rules using `Update` change types.

`ResaleAuthorization` must be available and targeted to you as a partner.

- **Name** (string) (optional) – Name associated with the offer for better readability to you and your customers. It will be displayed as part of Agreement information as well.

- **AgreementId** (string) (required) – AgreementId is the unique identifier of the agreement created when the targeted buyer accepted the previous offer you are trying to replace

Response Syntax

A change set is created for your request. The response to this request gives you the ChangeSetId and ChangeSetArn for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to CreateReplacementOfferUsingResaleAuthorization actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Name	Length must be between 1 and 150 characters
AgreementId	Provided agreement must be active Provided agreement must exist Provided agreement must be owned by Channel Partner
ResaleAuthorizationId	Required Length must be between 1 and 50 characters

Input field	Validation rule
ResaleAuthorizationId	ResaleAuthorization must be targeted to the channel partner.
ResaleAuthorizationId	ResaleAuthorization must be active
Channel Partner	Channel Partner must be paid seller in AWS Marketplace

Asynchronous Errors

The following errors are specific to `CreateReplacementOfferUsingResaleAuthorization` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_RESALE_AUTHORIZATION	Use a ResaleAuthorization in active state.
INCOMPATIBLE_RESALE_AUTHORIZATION	The ResaleAuthorization must be for the same product that is associated with the agreement.
INCOMPATIBLE_RESALE_AUTHORIZATION	Use a ResaleAuthorization targeted to the acceptor of the agreement.

Update markup

You can use the Catalog API to update pricing terms by a percentage value in your offer in AWS Marketplace.

This will apply the given percentage markup on all pricing terms and payment terms (for future payment schedules) that are defined by the manufacturer in the ResaleAuthorization. Any existing markup will be overwritten. You can view updated pricing and payment terms using `DescribeEntity`.

To update markup, call the `StartChangeSet` API operation with the `UpdateMarkup` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateMarkup",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Percentage": "5.0"
      }
    }
  ]
}
```

Provide information for the fields to add the `UpdateMarkup` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Percentage** (string) (required) – Percentage value will be added to the manufacturer pricing or payment terms.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
```



```
"ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information with the AWS Marketplace Seller Operations team to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdateMarkup actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Percentage	Required
	Data type is "String"
	Must be non-negative
	Allow up to 9 decimals

Asynchronous Errors

The following errors are specific to UpdateMarkup actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_MARKUP	The requested change can't be performed after the offer is released.

Error code	Error message
INCOMPATIBLE_RESALE_AUTHORIZATION	MarkupPercentage can't be updated when PaymentScheduleTerm or FixedUpfrontPricingTerm are present in offer and not present in ResaleAuthorization .
INVALID_MARKUP_PERCENTAGE	UpdateMarkup can only be invoked for offers created using ResaleAuthorization.

Update targeting configuration

You can use the Catalog API to replace the existing targeting configuration completely in AWS Marketplace.

Any existing targeting options that are not included in the latest request will be removed from the offer. Manufacturers can mention specific targeted buyers in ResaleAuthorization. Channel partners can give private offers to a subset of buyers using PositiveTargeting in the UpdateTargeting change type.

To update targeting configurations of your offer, call the StartChangeSet API operation with the UpdateTargeting change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "CountryCodes": [
```

```
        "US",
        "CA"
    ],
    "BuyerAccounts": [
        "111222333444"
    ]
},
"NegativeTargeting": {
    "CountryCodes": [
        "XX"
    ]
}
}
}
]
```

Provide information for the fields to add the `UpdateTargeting` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **PositiveTargeting** (object) (optional) – Positive targeting defines the criteria which any buyer's profile should fulfill in order to be allowed to access the offer. This field is optional, but at least one targeting option should be provided when this field is present.
 - **CountryCodes** (array of strings) (optional) – List as option for allowing targeting based on country. If the intention isn't to target the offer to a country, this field should be omitted. If it's present, the list must contain at least one country code. Each element in this list should be a valid 2-letter country code, using this format: ISO 3166-1 alpha-2.
 - **BuyerAccounts** (array of strings) (optional) – List as an option to allow targeting based on AWS accounts (also known as, Private Offer). If the intention is to not target the offer to an AWS account, this field should be omitted.
 - **NegativeTargeting** (object) (optional) – Negative targeting defines the criteria which any customer's profile should fulfill to be restricted to access the offer. Although this field is optional, at least one targeting option should be provided when this field is present.
 - **CountryCodes** (array of strings) (required) – List as option for allowing targeting based on country. If the intention isn't to target the offer to a specific country, then this field should

be omitted. If it's present, the list must contain at least one country code. Each element in this list should be a valid 2-letter country code using this format: ISO 3166-1 alpha-2.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateTargeting` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Details	Required	422
PositiveTargeting	Optional	422
NegativeTargeting	Optional	422
PositiveTargeting.CountryCodes	Optional Country codes must be valid (ISO 3166-1 alpha-2)	422
PositiveTargeting.BuyerAccounts	Optional	422

Input field	Validation rule	HTTP code
	AWS account IDs must be in valid format (12-digit number) Must not contain more than 25 accounts	
NegativeTargeting.CountryCodes	Optional Country codes must be valid (ISO 3166-1 alpha-2)	422
NegativeTargeting.BuyerAccounts	Must not be provided (negative targeting on BuyerAccounts isn't supported)	422

Asynchronous Errors

The following errors are specific to UpdateTargeting actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_BUYER_ACCOUNTS	Provide valid buyer accounts. Invalid accounts: [x].
INVALID_COUNTRY_CODES	Provide supported country codes.
INVALID_TARGETING	Use either negative or positive targeting on the same attribute.
INCOMPATIBLE_PRODUCT	Country-based targeting isn't supported for the product.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide BuyerAccounts that are compatible with the ResaleAuthorization.

Error code	Error message
INCOMPATIBLE_TARGETING	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TARGETING	The requested change can't be performed after the offer is expired.
INCOMPATIBLE_TARGETING	Targeting can't be updated on a replacement offer. If the buyer isn't associated with the provided AgreementId, then create a new private offer by providing an AgreementId associated with the buyer.
TOO_MANY_BUYER_ACCOUNTS	Provide BuyerAccounts within the allowed limits.

Update legal resources

You can use the Catalog API to merge the Resale Authorization legal terms and replace the existing legal terms completely in AWS Marketplace.

This change doesn't affect existing agreements. The legal terms that aren't included in the latest request will be removed from the offer. You can view the merged legal terms by calling `DescribeEntity`.

To update legal terms of your offer, call the `StartChangeSet` API operation with the `UpdateLegalTerms` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateLegalTerms",
```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "offer-123456789"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
            }
          ]
        }
      ]
    }
  ]
}

```

Provide information for the fields to add the `UpdateLegalTerms` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Terms** (array of structures) (required) – List of legal terms that you want to update. Supported legal terms are:
 - **LegalTerm** (object) (required) – Defines the list of text agreements to be proposed to the acceptors. One example of such an agreement is the end user license agreement (EULA).
 - **Type** (string) (required) – Category of term being updated.
 - **Documents** (array of structures) (required) – List of references to legal resources to be proposed to the buyers. One example of such a resource is the end user license agreement (EULA). Each reference is made up of a Type and a URL:
 - **Type** (string) (required) – Type of document. Available document types are:
 - **CustomEula** – A custom EULA provided by you as seller. A URL for a EULA stored in an accessible S3 bucket is required for this document type.

- **StandardEula** – Standard Contract For AWS Marketplace (SCMP). For more information about SCMP, see the AWS Marketplace Seller Guide. You don't provide a URL for this type because it is managed by AWS Marketplace.
- **Url** (string) (conditionally required) – A URL to the legal document for buyers to read. Required when Type is one of the following [CustomEula].
- **Version** (string) (conditionally required) – A version of standard contracts provided by AWS Marketplace. This is required when Type is StandardEula. Available versions are:
 - **2022-07-14** – This version of the Standard Contract for AWS Marketplace is available from this Amazon S3 bucket: <https://s3.amazonaws.com/aws-mp-standard-contracts/Standard-Contract-for-AWS-Marketplace-2022-07-14.pdf>

A change set is created for your request. The response to this request gives you the ID for the change set.

Response Syntax

```
{
  "ChangeSetId": "example123456789012abcdef", "ChangeSetArn": "arn:aws:aws-
  marketplace:us-east-
  1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information with the AWS Marketplace Seller Operations team to ensure it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action.

Synchronous Validations

The following schema validations are specific to UpdateLegalTerms actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	Required	422

Input field	Validation rule	HTTP code
	Only LegalTerm is allowed in the list List size must be 1	
Terms[].LegalTerm.Documents	Required	422
Terms[].LegalTerm.Documents[].Type	Required Allowed values: <ul style="list-style-type: none"> • CustomEula • StandardEula 	422
Terms[].LegalTerm.Documents[].Url	Required and must be a valid URL when "Type" is one of <ul style="list-style-type: none"> • CustomEula 	422

Asynchronous Errors

The following errors are specific to UpdateLegalTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_LEGAL_DOCUMENTS	Provide URLs for legal documents stored in accessible S3 buckets.

Error code	Error message
INVALID_LEGAL_DOCUMENTS	Only the most recent version of StandardEula is supported for new offers.
INVALID_LEGAL_DOCUMENTS	Provide legal documents in the supported file formats.
INVALID_LEGAL_DOCUMENTS	Provide legal documents using the supported document types.
LIMIT_EXCEEDED_LEGAL_DOCUMENT_SIZE	Provide legal documents within the allowed size limits.

Update the discoverability of the CPPO

You can use the Catalog API to manage the discoverability of your offer in AWS Marketplace. This change type doesn't affect existing agreements.

You can either choose to set a specific date in the future to restrict the discoverability of your offer or in the past to expire your offer.

To manage the discoverability of your offer, call the `StartChangeSet` API operation with the `UpdateAvailability` change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2024-05-31"
      }
    }
  ]
}
```

```
    }  
  }  
]  
}
```

Provide information for the fields to add the `UpdateAvailability` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **AvailabilityEndDate** (string) – Date until when the offer is discoverable and purchasable in AWS Marketplace. You can choose to set a specific date in the future to restrict the availability or in the past to expire the offer. Dates are represented in YYYY-MM-DD format. Offer expires at 23:59:59.999 UTC on the date provided.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{  
  "ChangeSetId": "example123456789012abcdef",  
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/  
ChangeSet/example123456789012abcdef"  
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateAvailability` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
AvailabilityEndDate	Required Format: "YYYY-MM-DD"	422

Asynchronous Errors

The following errors are specific to UpdateAvailability actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INVALID_AVAILABILITY_END_DATE	AvailabilityEndDate isn't supported for public offers.
INVALID_AVAILABILITY_END_DATE	Provide a future AvailabilityEndDate.
INVALID_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate that is before AgreementEndDate.
MISSING_AVAILABILITY_END_DATE	Provide an AvailabilityEndDate that is before the agreement's end date.

Define the expiration date of agreements

You can use the Catalog API to define the expiration date of the agreements that are created using this offer in AWS Marketplace.

This change does not affect existing agreements. The manufacturer could mention maximum agreement start date in a Resale Authorization. However, channel partners can't provide an agreement start date later than that date.

To define the expiration date of agreements, call the StartChangeSet API operation with the UpdateValidityTerms change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateValidityTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "ValidityTerm",
            "AgreementDuration": "P12M",
            "AgreementStartDate": "2021-08-01",
            "AgreementEndDate": "2022-08-01"
          }
        ]
      }
    }
  ]
}
```

Provide information for the fields to add the UpdateValidityTerms change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always Offer@1.0.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
 - **Terms** (array of structures) – List of validity terms that you want to update. Supported validity terms are:
 - **ValidityTerm** (object) – Defines the conditions that will keep an agreement, created from this offer, valid.
 - **Type** (string) – Category of the term being updated.

- **AgreementDuration** (string) – Defines the duration that the agreement remains active. If `AgreementStartDate` isn't provided, agreement duration is relative to the agreement signature time. The duration is represented in the ISO_8601 format.
- **AgreementStartDate** (string) – Defines the date when agreement starts. `AgreementStartDate` is represented in YYYY-MM-DD format. The agreement starts at 00:00:00.000 UTC on the date provided. If `AgreementStartDate` isn't provided, agreement start date is determined based on agreement signature time.
- **AgreementEndDate** (string) – Defines the date when the agreement ends. The `AgreementEndDate` is represented in YYYY-MM-DD format. The agreement ends at 23:59:59.999 UTC on the date provided. If `AgreementEndDate` isn't provided, the agreement end date is determined by the validity of individual terms.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdateValidityTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP code
Terms	<p>Required</p> <p>Only "ValidityTerm" is allowed in the list</p> <p>Must be empty or contain only 1 term</p>	422
Terms[].ValidityTerm	<p>Supported use cases:</p> <ol style="list-style-type: none"> 1. ValidityTerm with only Agreement Duration 2. ValidityTerm with only Agreement StartDate 3. ValidityTerm with only Agreement EndDate 4. ValidityTerm with both AgreementStartDate and Agreement EndDate 	422
Terms[].ValidityTerm.AgreementDuration	<p>Optional</p> <p>Represented in ISO_8601 format.</p>	422
Terms[].ValidityTerm.AgreementStartDate	<p>Optional</p> <p>Format: "YYYY-MM-DD"</p>	422
Terms[].ValidityTerm.AgreementEndDate	<p>Optional</p> <p>Format: "YYYY-MM-DD"</p>	422

Asynchronous Errors

The following errors are specific to `UpdateValidityTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_AGREEMENT	AgreementStartDate can't be in the future when the current agreement to be replaced isn't future dated.
INCOMPATIBLE_AGREEMENT_END_DATE	AgreementEndDate can't be updated after the offer is released.
INCOMPATIBLE_AGREEMENT_START_DATE	AgreementStartDate can't be updated after the offer is released.
INCOMPATIBLE_PRODUCT	AgreementStartDate in the future isn't supported.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the duration between Agreement StartDate and AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementStartDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the duration between Agreement StartDate and AgreementEndDate is compatible with the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure AgreementDuration matches duration specified in the ResaleAuthorization.
INCOMPATIBLE_TERMS	ValidityTerm isn't supported for public offers.

Error code	Error message
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_AGREEMENT_DURATION	Provide AgreementDuration that is greater than or equal to [x] days.
INVALID_AGREEMENT_END_DATE	Provide a future AgreementEndDate.
INVALID_AGREEMENT_END_DATE	Provide AgreementEndDate that is after or equal to [x].
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is after AvailabilityEndDate.
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is before the AgreementEndDate.
INVALID_AGREEMENT_START_DATE	Provide an AgreementStartDate that is within [x] years from today.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with both AgreementDuration and AgreementEndDate isn't supported.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with both AgreementStartDate and AgreementDuration isn't supported in an offer for the product.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with AgreementStartDate isn't supported in an offer for the product.
INVALID_AGREEMENT_TIME_INTERVAL	ValidityTerm with only AgreementStartDate isn't supported.
INVALID_AGREEMENT_TIME_INTERVAL	AgreementEndDate isn't supported unless it's used in combination with a future AgreementStartDate or for replacement offers.

Error code	Error message
INVALID_AGREEMENT_TIME_INTERVAL	Provide AgreementStartDate and AgreementEndDate where the difference is less than or equal to [x] years.
MISSING_AGREEMENT_START_DATE	Ensure AgreementStartDate is present in ValidityTerm when used along with ConfigurableUpfrontPricingTerm.

Update pricing

You can use the Catalog API to replace the existing pricing terms completely.

The pricing terms that aren't included in the latest request will be removed from the offer. Channel partners can use this change type only to pass FixedUpfrontPricingTerm.

To update pricing terms for your offers, call the StartChangeSet API operation with the UpdatePricingTerms change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "Details": {
        "PricingModel": "Contract",
        "Terms": [
          {
            "Type": "FixedUpfrontPricingTerm",
            "CurrencyCode": "USD",
            "Price": "200.00",
```

```

        "Duration": "P465D",
        "Grants": [
            {
                "DimensionKey": "Users",
                "MaxQuantity": 10
            }
        ]
    }
]
}
}
}
}

```

Provide information for the fields to add the `UpdatePricingTerms` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **Details** (string) (required) – Specifics of the request. This field is a JSON string field. It must be formatted properly for a single-line string field, including escaping characters (such as quotation marks) that can't be in a string.
- **PricingModel** (string) (required) – Pricing model for your offer. Possible values for pricing model are:
 - **Contract** – Contract-based pricing model where buyers are either billed in advance for the use of your product, or offered a flexible payment schedule. Buyers can also pay for an additional usage above their contract.
 - **Terms** (array of structures) (required) – List of pricing terms that you want to update. Supported pricing terms are:
 - **FixedUpfrontPricingTerm** (object) – Defines a pre-paid pricing model where the customers are charged a fixed upfront amount.
 - **Type** (string) (required) – Type of the term being updated.
 - **CurrencyCode** (string) (required) – Defines the currency for the prices mentioned in this term. For public offers, only USD is supported. For private offers, USD, AUD, EUR, GBP, and JPY are supported.
 - **Price** (string) (required) – Fixed amount to be charged to the customer when this term is accepted.

- **Grants** (array of structures) (required) – Entitlements that will be granted to the acceptor of fixed upfront as part of agreement execution.
- **DimensionKey** (string) (required) – Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.
- **MaxQuantity** (integer) (optional) – Maximum amount of capacity that the buyer can be entitled to the given dimension of the product. If `MaxQuantity` is not provided, the buyer will be able to use an unlimited amount of the given dimension.
- **Duration** (string) (optional) – Defines the duration that the term remains active. This field supports the ISO 8601 format.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `UpdatePricingTerms` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
<code>PricingModel</code>	Required

Input field	Validation rule
	Allowed pricing models: Contract
Terms	Required Allowed terms: FixedUpfrontPricingTerm
Terms[].FixedUpfrontPricingTerm.CurrencyCode	Required Allowed values: ["USD", "AUD", "EUR", "GBP", "JPN"] Allowed pricing models: Contract
Terms[].FixedUpfrontPricingTerm.Price	Required Data type is "String" Must be non-negative Support up to 6 Decimals No special character supported
Terms[].FixedUpfrontPricingTerm.Duration	Required Expected format: ISO 8601 duration
Terms[].FixedUpfrontPricingTerm.Grants[].DimensionKey	Required Length must be between 1 and 60
Terms[].FixedUpfrontPricingTerm.Grants[].MaxQuantity	Required

Asynchronous Errors

The following errors are specific to `UpdatePricingTerms` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
DUPLICATE_DIMENSION_KEYS	Provide Grants with a unique list of dimension keys in [x].
DUPLICATE_DIMENSION_KEYS	Provide RateCard with a unique list of dimension keys in [x].
DUPLICATE_SELECTORS	Provide a unique list of Selectors in ConfigurableUpfrontPricingTerm.
DUPLICATE_TERM_TYPES	Provide a unique list of term types.
INCOMPATIBLE_AGREEMENT	The following terms can't be removed from the replacement offer: [x, y, z].
INCOMPATIBLE_AGREEMENT	The following terms can't be added to the replacement offer: [x, y, z].
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_PRODUCT	Usage pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	Contract pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	Byol pricing model isn't supported for the product.

Error code	Error message
INCOMPATIBLE_PRODUCT	Free pricing model isn't supported for the product.
INCOMPATIBLE_PRODUCT	[x] isn't supported in an offer for the product.
INCOMPATIBLE_PRODUCT	Provided payment and pricing terms are incompatible.
INCOMPATIBLE_PRODUCT	Use existing, available dimensions in the product in [x].
INCOMPATIBLE_PRODUCT	FreeTrialPricingTerm as the offer's only pricing term isn't supported for the product.
INCOMPATIBLE_PRODUCT	The following terms aren't supported for the product: [x,y,z].
INCOMPATIBLE_PRODUCT	Replacement offers are only supported for contract pricing model.
INCOMPATIBLE_PRODUCT	Provide pricing term(s) that are compatible with the product dimensions. Incompatible pricing terms: [x,y,z].
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	Set MultipleDimensionSelection and QuantityConfiguration to Allowed in ConfigurableUpfrontPricingTerm for usage pricing model.
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	Set MultipleDimensionSelection and QuantityConfiguration to Disallowed in ConfigurableUpfrontPricingTerm for usage pricing model.
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	QuantityConfiguration in ConfigurableUpfrontPricingTerm can't be changed after the offer is released.

Error code	Error message
INCOMPATIBLE_RATE_CARD_CONSTRAINTS	MultipleDimensionSelection in ConfigurableUpfrontPricingTerm can't be changed after the offer is released.
INCOMPATIBLE_RATES	Set all charge amounts and prices to zero (0) when using Free pricing model.
INCOMPATIBLE_RATES	Only zero (0) prices are allowed in UsageBasedPricingTerm for a free trial offer for the product.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide the same CurrencyCode that is specified in the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure Duration in FixedUpfrontPricingTerm matches duration specified in the ResaleAuthorization.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide term(s) that are compatible with the ResaleAuthorization. Incompatible terms: [x, y, z].
INCOMPATIBLE_SELECTOR_DURATION	Durations aren't allowed to be removed from rate cards in ConfigurableUpfrontPricingTerm after the offer released.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INCOMPATIBLE_TERMS	[x] isn't supported together with the following terms: [y,z].
INCOMPATIBLE_TERMS	The following terms can't be added after the offer is released: [x,y,z].
INCOMPATIBLE_TERMS	The following terms can't be removed after the offer is released: [x,y,z].

Error code	Error message
INCOMPATIBLE_TERMS	[x] isn't supported for private offers.
INCOMPATIBLE_TERMS	The following terms aren't supported with FreeTrialPricingTerm that grants unlimited usage: [x,y,z].
INCOMPATIBLE_TERMS	The following terms aren't supported with FreeTrialPricingTerm for the product: [x,y,z].
INCOMPATIBLE_TERMS	Provide zero (0) price for FixedUpfrontPricingTerm when the offer contains a PaymentScheduleTerm.
INCOMPATIBLE_TERMS	The following terms aren't compatible with the PricingModel: [x,y,z].
INCOMPATIBLE_TERMS	FixedUpfrontPricingTerm isn't supported when MarkupPercentage is greater than zero (0).
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_AGREEMENT_DURATION	Provide duration between [x] and [y] months.
INVALID_AGREEMENT_DURATION	Ensure duration granularity is at the day level for metered dimensions.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
INVALID_DURATION	Ensure Duration in FreeTrialPricingTerm is within the allowed range.

Error code	Error message
INVALID_DURATION	Provide Duration in FixedUpfrontPricingTerm that matches the duration between AgreementStartDate and AgreementEndDate.
INVALID_DURATION	Provide duration between [x] and [y] months.
INVALID_DURATION	Ensure duration granularity is at the day level for metered dimensions.
INVALID_GRANTS	Provide the same MaxQuantity for all Grants in FreeTrialPricingTerm.
INVALID_GRANTS	Provide Grants for all available metered dimensions in FreeTrialPricingTerm.
INVALID_PRICE_CHANGE	[x] can't be updated until [y] because you have requested a price increase in the past 120 days. To cancel your previous price increase request or for more information, contact the AWS Marketplace Managed Catalog Operations Team.
INVALID_PRICE_CHANGE	Price increase and dimension addition in [x] isn't supported in the same request. Add dimensions first.
INVALID_PRICE_CHANGE	Price increase and decrease in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.
INVALID_PRICE_CHANGE	Price increase in RecurringPaymentTerm and price decrease in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.

Error code	Error message
INVALID_PRICE_CHANGE	Price decrease in RecurringPaymentTerm and price increase in UsageBasedPricingTerm isn't supported in the same request. Decrease prices first.
INVALID_RATE_CARD	ConfigurableUpfrontPricingTerm is missing one or more-dimension keys for duration [x]. Provide prices for the same set of dimension keys for all durations.
INVALID_RATE_CARD	Provide a rate card for only metered dimensions in UsageBasedPricingTerm.
INVALID_RATE_CARD	Rates can't be removed from [x]. Provide prices for all dimensions in the existing rate card.
INVALID_RATE_CARD	Provide dimensions that have the same unit in [x].
INVALID_RATE_CARD	Provide either all metered or all entitled dimensions in [x].
INVALID_RATE_CARD	Provide only entitled dimensions in [x].
INVALID_RATE_CARD	Provide usage based rates for all available metered dimensions in UsageBasedPricingTerm.
INVALID_RATE_CARD	Provide usage based rates for all free trial dimensions.
INVALID_RATE_CARD	Provide prices with up to 3 decimal places in UsageBasedPricingTerm.
INVALID_SELECTOR_DURATION_VALUE	Provide duration between [x] and [y] months.

Error code	Error message
INVALID_SELECTOR_DURATION_VALUE	Ensure duration granularity is at the day level for metered dimensions.
INVALID_SELECTOR_DURATION_VALUE	Ensure Duration in ConfigurableUpfrontPricingTerm is within the allowed range.
INVALID_SELECTOR_DURATION_VALUE	Provide one or more supported contract durations.
INVALID_SELECTOR_DURATION_VALUE	Provide one or more supported contract durations or a single custom duration.
INVALID_SELECTOR_DURATION_VALUE	Provide Duration in ConfigurableUpfrontPricingTerm that matches the duration between AgreementStartDate and AgreementEndDate.
MISSING_DURATION	Provide Duration in FixedUpfrontPricingTerm.
MISSING_MANDATORY_TERMS	FixedUpfrontPricingTerm is only supported when paired with ByolPricingTerm or PaymentScheduleTerm.
MISSING_MANDATORY_TERMS	Provide at least one of [x,y,z].
MISSING_MANDATORY_TERMS	Provide a ByolPricingTerm when using Byol pricing model.
TOO_MANY_GRANTS	Provide up to [x] grants in [y].
TOO_MANY_RATE_CARDS	Only one rate card in ConfigurableUpfrontPricingTerm is allowed for the product.
TOO_MANY_RATE_CARDS	Up to [x] rate cards are allowed in ConfigurableUpfrontPricingTerm for the product.
TOO_MANY_RATES	Provide RateCards within the allowed limits in ConfigurableUpfrontPricingTerm.

Error code	Error message
TOO_MANY_RATES	Provide RateCards within the allowed limits in UsageBasedPricingTerm.

Update payment schedule details

You can use the Catalog API to change the payment schedule details, such as flexible payment schedule, in AWS Marketplace.

Channel partners can't override the payment schedule terms if the manufacturer provided terms in the Resale Authorization. You can increase the payment amount using the UpdateMarkup change type.

To update payment-associated details for your offer, call the StartChangeSet API operation with the UpdatePaymentScheduleTerms change type, as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePaymentScheduleTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-123456789"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "PaymentScheduleTerm",
            "Schedule": [
              {
                "ChargeDate": "2021-12-01",
                "ChargeAmount": "200.00"
              },
              {
```

```

        "ChargeDate": "2022-03-01",
        "ChargeAmount": "250.00"
    }
  ]
}

```

Provide information for the fields to add the `UpdatePaymentScheduleTerms` change type:

- **Entity** (object) (required) – Your CPPO.
- **Type** (string) (required) – The Type is always `Offer@1.0`.
- **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request.
- **Terms** (array of structures) – List of payment terms that you want to update. Supported payment terms are:
 - **PaymentScheduleTerm** (object) – Defines an installment-based pricing model where customers are charged a fixed price on different dates during the agreement validity period.
 - **Type** (string) – Type of the term being updated. This is the object value: `"PaymentScheduleTerm"`.
 - **Schedule** (array of structures) – List of the payment schedule where each element defines one installment of payment. It contains the information necessary for calculating the price to be paid and the date on which the customer would be charged.
 - **ChargeDate** (string) – The date on which the customer would pay the price defined in this payment schedule term. `ChargeDate` is represented in YYYY-MM-DD format. Invoices are generated on the date provided.
 - **ChargeAmount** (string) – The price that the customer would pay on scheduled date (`ChargeDate`).

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to UpdatePaymentScheduleTerms actions in the AWS Marketplace Catalog API. These validations are performed when you call StartChangeSet. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule	HTTP
Terms	Required Only "PaymentScheduleTerm" is allowed List size must be less than 2	422
Terms[].PaymentScheduleTerm. .CurrencyCode	Required Supported currencies: ["USD", "AUD", "EUR", "GBP", "JPN"]	422
Terms[].PaymentScheduleTerm. .Schedule[]	Required List size must be between 1 and 60, inclusive	422
Terms[].PaymentScheduleTerm. .Schedule[].ChargeDate	Required Format: "YYYY-MM-DD"	422

Input field	Validation rule	HTTP
Terms[].PaymentScheduleTerm .Schedule[].ChargeAmount	Required Data type is "String" Non-negative decimals with up to 2 decimal places supported No additional properties are allowed	422

Asynchronous Errors

The following errors are specific to UpdatePaymentScheduleTerms actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more information about using DescribeChangeSet to get the status of a change request, see [Working with change sets](#).

Error code	Error message
DUPLICATE_CHARGE_DATES	Provide unique charge dates in PaymentScheduleTerm.
INCOMPATIBLE_CURRENCY_CODE	CurrencyCode can't be changed after the offer is released.
INCOMPATIBLE_MARKUP_PERCENTAGE	PaymentScheduleTerm isn't supported when MarkupPercentage is greater than zero (0).
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the CurrencyCode.
INCOMPATIBLE_RESALE_AUTHORIZATION	Provide term(s) that are compatible with the ResaleAuthorization. Incompatible terms: [PaymentScheduleTerm].
INCOMPATIBLE_RESALE_AUTHORIZATION	Ensure the total ChargeAmounts in PaymentScheduleTerm is compatible with the ResaleAuthorization.

Error code	Error message
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is released.
INCOMPATIBLE_TERMS	The requested change can't be performed after the offer is expired.
INVALID_CHARGE_DATES	Provide charge dates before Agreement EndDate.
INVALID_CURRENCY_CODE	Provide a supported CurrencyCode.
INVALID_CURRENCY_CODE	Provide the same CurrencyCode across all pricing and payment terms.
TOO_MANY_BACKDATED_CHARGES	Provide up to 1 scheduled payment before AvailabilityEndDate.

Publish the CPPO

You can use the Catalog API to merge the information collected from all update change types, and then publish the offer in AWS Marketplace.

Offers remain in a `Draft` state, until `ReleaseOffer` is called. After the offer is released, it's discoverable in AWS Marketplace.

To publish your offer, call the `StartChangeSet` API operation with the `ReleaseOffer` change type as shown in the following example.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
```

```
"ChangeSet": [  
  {  
    "ChangeType": "ReleaseOffer",  
    "Entity": {  
      "Type": "Offer@1.0",  
      "Identifier": "offer-123456789"  
    },  
    "DetailsDocument": {}  
  }  
]
```

Provide information for the fields to add the `ReleaseOffer` change type:

- **Entity** (object) (required) – Your CPPO.
 - **Type** (string) (required) – The Type is always `Offer@1.0`.
 - **Identifier** (string) (required) – Your offer ID. For more information, see [Identifier](#).
- **DetailsDocument** (object) (required) – The JSON value of specifics of the request. It must be empty for `ReleaseOffer`.

Response Syntax

A change set is created for your request. The response to this request gives you the `ChangeSetId` and `ChangeSetArn` for the change set and looks like the following.

```
{  
  "ChangeSetId": "example123456789012abcdef",  
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/  
ChangeSet/example123456789012abcdef"  
}
```

The change request is added to a queue and processed. This includes validating information to ensure that it meets the AWS Marketplace guidelines. The validation process can take anywhere from a few minutes to a few hours.

You can check the status of the request through the AWS Marketplace Management Portal, or directly through Catalog API using the [DescribeChangeSet](#) API operation.

Synchronous Validations

The following schema validations are specific to `ReleaseOffer` actions in the AWS Marketplace Catalog API. These validations are performed when you call `StartChangeSet`. If the request doesn't meet the following requirements, it will fail with an HTTP response.

Input field	Validation rule
Details	Must be empty ({})

Asynchronous Errors

The following errors are specific to `ReleaseOffer` actions in the AWS Marketplace Catalog API. These errors are returned when you call `DescribeChangeSet` after a change set is processing. For more information about using `DescribeChangeSet` to get the status of a change request, see [Working with change sets](#).

Error code	Error message
INCOMPATIBLE_PAYMENT_SETTINGS	Update your payment settings to be compatible with the <code>CurrencyCode</code> .
INCOMPATIBLE_PRODUCT	First create a public offer for the product.
INCOMPATIBLE_SELLER_VERIFICATION	Complete all required seller verification processes.
INVALID_UPDATE_REQUEST	The requested change can't be performed after the offer is released.
MISSING_AGREEMENT_END_DATE	Provide an <code>AgreementEndDate</code> for replacement offers.
MISSING_AVAILABILITY_END_DATE	Provide an <code>AvailabilityEndDate</code> for private offer.
MISSING_MANDATORY_TERMS	Provide a <code>FixedUpfrontPricingTerm</code> when the offer contains a <code>PaymentScheduleTerm</code> .

Error code	Error message
MISSING_BUYER_ACCOUNTS	Provide PositiveTargeting with BuyersAccounts for offers created using ResaleAuthorization.
MISSING_BUYER_ACCOUNTS	All offers for the product must be private. Provide PositiveTargeting with BuyersAccounts.
MISSING_DESCRIPTION	Set Description before releasing the offer.
MISSING_MANDATORY_TERMS	Add [x] to the offer.
MISSING_MANDATORY_TERMS	Provide a FixedUpfrontPricingTerm when the offer contains a PaymentScheduleTerm.
MISSING_NAME	Set Name before releasing the offer.
TOO_MANY_OFFERS	Only one public offer can be created per product.
TOO_MANY_OFFERS	Only one public free trial offer can be created per product.

Define an existing CPPO

You can use the Catalog API to define CPPO details in AWS Marketplace.

To define an existing CPPO, call the DescribeEntity API operation with the Offer@1.0 entity type, as shown in the following example.

Request Syntax

```
GET /DescribeEntity?catalog=<Catalog>&entityId=<EntityId> HTTP/1.1
```

Provide information for the fields to add the DescribeEntity change type:

- **catalog** (string) –The catalog related to the request. Fixed value: AWSMarketplace.

- **entityId** (string) – The unique ID of the offer to describe.

Response Syntax

The response to this request gives you the offer details and looks like the following.

```
{
  "EntityType": "Offer@1.0",
  "EntityIdentifier": "offer-a5oEXAMPLEzpu@1",
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:446235747164:AWSMarketplace/Offer/offer-a5oEXAMPLEzpu",
  "LastModifiedDate": "2021-03-10T21:57:16Z",
  "Details": {
    "Id": "offer-3rEXAMPLErn",
    "State": "Released",
    "Name": "Test Offer",
    "Description": "Worldwide private offer for Test Product",
    "PreExistingAgreement": {
      "AcquisitionChannel": "External",
      "PricingModel": "Contract"
    },
    "ProductId": "prod-0bEXAMPLEb51",
    "ResaleAuthorizationId": "resaleauthz-123456789",
    "MarkupPercentage": "5.0",
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
          }
        ]
      }
    ],
    {
      "Type": "ConfigurableUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P12M"
          }
        }
      ],
    }
  ]
}
```

```
    "RateCard": [
      {
        "DimensionKey": "m3.large",
        "Price": "300.00"
      },
      {
        "DimensionKey": "m4.xlarge",
        "Price": "400.00"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
],
{
  "Type": "UsageBasedPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "RateCard": [
        {
          "DimensionKey": "m3.large",
          "Price": "0.10"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "0.20"
        }
      ]
    }
  ]
},
{
  "Type": "PaymentScheduleTerm",
  "CurrencyCode": "USD",
  "Schedule": [
    {
      "ChargeDate": "2020-12-01T00:00:00Z",
      "ChargeAmount": "1000.00"
    },
    {
```

```

        "ChargeDate": "2021-06-15T00:00:00Z",
        "ChargeAmount": "1250.00"
    }
]
},
"Rules": [
    {
        "Type": "TargetingRule",
        "PositiveTargeting": {
            "CountryCodes": [
                "US",
                "CA"
            ],
            "BuyerAccounts": [
                "118033953248"
            ]
        },
        "NegativeTargeting": {
            "CountryCodes": [
                "XX"
            ]
        }
    },
    {
        "Type": "AvailabilityRule",
        "AvailabilityEndDate": "2050-08-30T01:56:03Z"
    }
]
}
}

```

Provide information for the input fields for the DescribeEntity response:

- **EntityType** (string) – The named type of the entity, which is Offer@1.0.
- **EntityIdentifier** (string) – The identifier of the entity, in the format of EntityId@RevisionId.
- **EntityArn** (string) – The ARN associated to the unique identifier for the change set referenced in this request.
- **LastModifiedDate** (string) – The last modified date of the entity, in ISO 8601 format (2018-02-27T13:45:22Z).
- **DetailsDocument** (object) – The JSON object includes the details of the entity.

- **Id** (string) – Unique identifier for an offer entity in AWS Marketplace and is generated during the creation of an offer.
- **State** (string) – The status of the offer.
- **Name** (string) – Name associated with the offer for better readability to you and your customers. It will be displayed as part of Agreement information as well.
- **Description** (string) – Description is a free-form text which is meant to be used only by you and will never be exposed to buyers.
- **PreExistingAgreement** (string) – Determines if this offer is a renewal for an existing agreement with an existing customer for the same underlying product. The existing agreement can be within or outside AWS Marketplace. AWS may audit and verify your offer is a renewal. If AWS is unable to verify your offer, then AWS may revoke the offer and entitlements from your customer.
 - **AcquisitionChannel** (string) – Indicates if the existing agreement was signed outside AWS Marketplace or within AWS Marketplace. Possible values: `External`, `AwsMarketplace`.
 - **PricingModel** (string) – Indicates which pricing model the existing agreement uses. Possible values: `Contract`, `Usage`, `ByoI`, `Free`.
- **ProductId** (string) – Description is a free-form text which is meant to be used only by you and will never be exposed to buyers.
- **ResaleAuthorizationId** (string) – ResaleAuthorization is used to create the private offer.
- **MarkupPercentage** (string) – Percentage value that the channel partner passed in the `UpdateMarkup` change type. This markup is already applied to the terms.
- **Terms** (array of structures) – List of terms.
- **Rules** (array of structures) – List of rules.

Work with renewals

Renewals in AWS Marketplace enable sellers, including independent software vendors (ISVs) and channel partners, to seamlessly extend existing agreements with customers. During the renewal process, sellers have the option to adjust terms by creating a new offer.

There are four types of renewals in AWS Marketplace:

- **Agreement based offers:** Sellers can replace an existing agreement to adjust pricing, duration, terms, and renew an existing contract before it ends. An agreement based offer can be extended

to the customer beyond the current agreement's end date, granting new entitlements, discounts, or payment schedules based on the customer's needs.

This support applies to software as a service (SaaS) products, including those with contract and consumption-based pricing (CCP), whether they offer flexible payment options or not.

- **Future dated offers:** Sellers can create future dated offers to start on a date in the future, and use them to pre-book renewals while existing terms are still ongoing. Once accepted by the customer, the agreement begins on a specified **future date, allowing to start after the current agreement ends.**

This support applies to software as a service (SaaS) products, including those with contract and consumption-based pricing (CCP), whether they offer flexible payment options or not.

- **New private offer:** Sellers can [create a new private offer](#) with new terms (for AMI hourly, AMI annual, and SaaS pay-as-you-go subscriptions) that can be accepted anytime to renew an existing agreement.
- **Auto-renewal:** Customers can enable auto-renewal for public offers and AWS Data Exchange products with private offers to automatically create a new agreement when a previous agreement ends.

Topics

- [Replacement offers](#)
- [Future dated agreements](#)
- [Resources](#)

Replacement offers

As a seller, you can offer renewals by replacing an active agreement that was originally created when the customer accepted your public offer or private offer. Using a replacement offer, you can extend a new offer to the customer that goes beyond the current agreement's end date, grant new entitlements, offer pricing discounts, adjust payment schedules, change the payment schedule, or change the end user license agreement (EULA).

You can use the Catalog API to [create a replacement offer](#) (also known as an agreement-based offer) in AWS Marketplace for [supported product types](#). You will need to provide the unique identifier (agreement ID) of the current agreement you wish to replace. You can find this

agreement ID in the [AWS Marketplace Management Portal](#) under the Agreements section or by using the [SearchAgreements](#) API operation.

Once the customer accepts the replacement offer, their current agreement will be replaced with a new agreement that can extend beyond the previous end date.

Note

You cannot create a replacement agreement that specifies a seller of record that differs from the original agreement.

For more information on replacement offers, see [Amending private offers in AWS Marketplace](#) in the *AWS Marketplace Seller Guide*.

Future dated agreements

Future-dated agreements are created when a customer accepts a private offer with a future service start date. To facilitate advance booking of upcoming renewals in AWS Marketplace, ISVs and channel partners can create private offers that start the day after the current agreement ends rather than immediately upon acceptance.

To set the start date of a renewal agreement as the day after the end date of the current agreement, you can use the [Catalog API UpdateValidityTerms change type](#) on a private offer that has already been published. Sellers can choose a service start date up to three years in the future.

Customers can review the terms and conditions of the private offer and accept it before it takes effect. Accepting a private offer with a start date in the future does not replace the current agreement. Instead, it creates a renewal agreement that begins immediately after the previous agreement ends.

For more information on product types that support future dated offers and agreements, see [Creating future dated agreements](#) in the *AWS Marketplace Seller Guide*.

Resources

- For end-to-end labs with working code examples, see:
 - [Lab: Create a private offer \(with a future service start date\)](#)
 - [Lab: Create a replacement private offer](#)

- For a video on creating replacement offers, see [Renew SaaS Contract Private Offers - AWS Marketplace](#) on YouTube.
- For a video on creating future dated offers, see [Create an AWS Marketplace Future Dated Private Offer](#) on YouTube.

Using the AWS Marketplace API as a buyer

The following sections provide information on how to use the API as a buyer.

Topics

- [Work with the catalog to discover products using the AWS Marketplace API](#)
- [Work with the private marketplace using the AWS Marketplace API](#)

Work with the catalog to discover products using the AWS Marketplace API

The Discovery API is currently available to select AWS customers and is available upon request to, and approval by, the AWS Marketplace team. For more information, see [Access control for the AWS Marketplace Discovery API](#).

Work with the private marketplace using the AWS Marketplace API

You can use the AWS Marketplace Catalog API to manage a *private marketplace* for your AWS account or [organization](#).

All change types can be called only from the organization's management account or by a member account that is a delegated administrator for private marketplace. If you're a current private marketplace customer without the AWS Organizations integration for private marketplace, you can create and manage a private marketplace from any account in your organization that has the `AWSPprivateMarketplaceAdminFullAccess` IAM policy.

For more information about private marketplaces, see [Private marketplaces](#) in the *AWS Marketplace Buyer Guide*.

The following table details a set of tasks to manage private marketplaces and the change types that apply to each task.

Task	Action	Change types
the section called "Creating a private marketplace"	StartChangeSet	CreateExperience CreateProcurementPolicy
the section called "Changing the branding of a private marketplace experience"	StartChangeSet	CreateBrandingSettings UpdateBrandingSettings
the section called "Enabling or disabling a private marketplace experience"	StartChangeSet	UpdateExperience
the section called "Enabling or disabling user requests"	StartChangeSet	UpdateProcurementPolicy
the section called "Getting a list of products in a private marketplace experience"	DescribeEntity	Not applicable
the section called "Adding or removing products from a private marketplace"	StartChangeSet	AllowProductProcurement DenyProductProcurement
the section called "Finding products"	Not applicable	Not applicable
the section called "Working with private marketplaces for AWS Organizations"	Not applicable	Not applicable
the section called "Associating principals to experiences"	StartChangeSet	AssociateAudience

Task	Action	Change types
		DisassociateAudience
Archiving and reactivating a private marketplace experience	StartChangeSet	RestrictExperience ReviveExperience

Creating a private marketplace

A private marketplace for an AWS account can be thought of as a list of products that users are allowed to procure in that account, and branding for the marketplace. In an organization with multiple accounts, you can use the grouping from AWS Organizations called [organizational unit \(OU\)](#) to associate to an experience. For example, you could have one set of products that all accounts in the organization are allowed to procure, or you could have a different list of products for each OU in the organization. You can also have a different list of products for individual accounts in the organization. Each list of approved products and branding is called a *procurement experience*.

In the AWS Marketplace Catalog API, four entities represent an experience:

- Experience entity – This entity is at the highest level of the experience and contains two child entities.
- ProcurementPolicy entity – This entity represents the products that have been allowed and denied in your private marketplace.
- BrandingSettings entity – You can also create a BrandingSettings entity to define how your private marketplace looks to your users.
- Audience entity – You must also associate one or more Audience entities, which define the set of AWS accounts, OUs, or organization that the experience applies to.

The steps to create a procurement experience are as follows:

1. Create the Experience entity.
2. Create a ProcurementPolicy entity to store the list of products that are allowed or denied for the experience.

3. (Optional) Create a `BrandingSettings` entity to customize the look of your marketplace experience.
4. Associate principals with your experience. A principal can be an AWS account, OU, or the organization.
5. Enable the experience.

Note

If your account is part of an organization in AWS Organizations, see [Working with private marketplaces for AWS Organizations](#).

Create the Experience entity

To create the Experience entity, use the `StartChangeSet` action with the `CreateExperience` value for the `ChangeType` parameter to request that the experience be created by AWS Marketplace. See the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "CreateExperience",
      "DetailsDocument":
      {
        "Name": "ExamplePrivateMarketplace"
      },
      "Entity":
      {
        "Type": "Experience@1.0"
      }
    }
  ],
  "ChangeSetName": "Create Private Marketplace Example"
}
```

In this action, `Entity` is a template for the entity that you want to create. It is assigned an `EntityId` when it is created. `ChangeSetName` identifies the change to help you find it later.

The response looks like the following.

```
{
  "ChangeSetArn": "arn:...:AWSMarketplace/ChangeSet/abcd1234example5678frjzkz",
  "ChangeSetId": "abcd1234example5678frjzkz"
}
```

The response includes a `ChangeSetId` that you can use to get the status of your change request as it is processed with `DescribeChangeSet`. You can also use `ListEntities` to find your `Experience` entity without the `ChangeSetId`. For more information about change sets, see [Working with change sets](#).

A newly created `Experience` entity doesn't have a procurement policy by default. It is also created with default settings for branding. For more information about branding settings, including how to customize them, see [Changing the branding of a private marketplace experience](#).

Create a `ProcurementPolicy` entity

You must create a `ProcurementPolicy` entity. By default, a new `Experience` entity is disabled, so you can create the procurement policy before enabling it.

Note

An `Experience` entity with *no* procurement policy (null) allows all products to be procured in your private marketplace. An `Experience` entity with an *empty* procurement policy has no products available to users to procure.

To allow and deny products in your private marketplace, you must create the procurement policy. To do this, you again call `StartChangeSet`, but this time with the `ChangeType` of `CreateProcurementPolicy`. The following code example creates an empty procurement policy.


```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
```



```
"Catalog": "AWSMarketplace",
"ChangeSet":
[
  {
    "ChangeType": "CreateProcurementPolicy",
    "DetailsDocument":
    {
      "Name": "ExampleProcurementPolicy"
    },
    "Entity":
    {
      "Type": "Experience@1.0",
      "Identifier": "exp-1234example@1"
    }
  }
]
```

The Entity you provide in this action is the Experience entity that you want the procurement policy created within, so you must include the identifier for the entity that you created earlier. Use `ListEntities` to find the Experience entity. You can also return the identifier by using `DescribeChangeSet` with the change set identifier from the `CreateExperience` action

 **Note**

This example shows the identifier with a revision of 1. For more information about revisions for identifiers, see [Identifier](#).

You can again use `DescribeChangeSet` on the `CreateProcurementPolicy` change type to follow the processing of your request.

 **Note**

The names you give the Experience and ProcurementPolicy objects do not appear in AWS Marketplace. The names are only for your ease of finding the entities in the API.

After you have created the procurement policy, your private marketplace displays in the AWS Management Console. (You can go to the [Private Marketplace page](#) to see it.) After you have

completed these steps, your private marketplace will be disabled, have default branding, have an empty procurement policy, and will not be associated with any principals in your organization. You can update the branding and add any products that you want in it, associate the experience with one or more accounts, and then enable your private marketplace.

The following sections describing managing your private marketplace with the AWS Marketplace Catalog API.

Changing the branding of a private marketplace experience

You can customize the look of your private marketplace for your users. Without customization, your private marketplace will have the default branding settings, which are described below. Aspects of branding that you can change in a private marketplace include the following:

- **Title** – The name displayed for your private marketplace. This is the same as the **Name** field in the private marketplace **Profile settings** screen. If you set the **Title** to **Example**, then the text displayed is **Example Private Marketplace**. The default is **Private Marketplace**.
- **Information** – The paragraph displayed under the name in your private marketplace. This is the same as the **Description** field in **Profile settings**. The default is no information, in which case a general description of private marketplaces is displayed.
- **ThemeColor** – The color displayed in the banner of your private marketplace. This is a color in RGB hexadecimal format. This value is the same as the **Theme color** field in **Profile settings**. The default value is #232F3E.
- **LogoUrl** – The URL that points to an image file to be used as the logo on your private marketplace. The URL must be publicly available (for example, a signed Amazon S3 URL). The file must be either a .png or .svg file and be under 500kb. If necessary, the image file will be resized to a maximum height of 30 pixels and a maximum width of 100 pixels. This is the same value as the **Logo Select** in **Profile Settings**. The default is to not show a logo.

To set these values, you must first create a `BrandingSettings` entity with the `CreateBrandingSettings` change type. You can then request an `UpdateBrandingSettings` change to set or change the branding. You only need to create a `BrandingSettings` object once. To create this object, call `StartChangeSet` with the `CreateBrandingSettings` change type, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "CreateBrandingSettings",
      "DetailsDocument":
      {
        "Name": "ExampleBrandingSettingsName"
      },
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@2"
      }
    }
  ]
}
```

This example modifies the Experience entity by adding the BrandingSettings object to it. The revision of the entity identifier has incremented to 2. For more information about revisions for identifiers, see [Identifier](#).

Note

You can specify all the details of the branding settings in the call to create the branding settings entity. The details facet is the same for CreateBrandingSettings and UpdateBrandingSettings.

You modify the settings by calling StartChangeSet with the UpdateBrandingSettings change type. The settings are part of the Configuration of the DetailsDocument object.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
```

```
"ChangeType": "UpdateBrandingSettings",
"DetailsDocument":
{
  "Name": "ExampleBrandingSettingsName",
  "Description": "Example description",
  "Configuration":
  {
    "Title": "ExampleName",
    "Information": "Example description.",
    "ThemeColor": "#0e7f74",
    "LogoUrl": "https://example.com/path/mylogo.png"
  }
},
"Entity":
{
  "Type": "Experience@1.0",
  "Identifier": "exp-1234example@3"
}
}
]
```

Note

The URL for the logo is used to make a copy during the update change. After the change is complete, if you remove or change the URL at that path, it will not affect your private marketplace unless you again request `UpdateBrandingSettings`.

Enabling or disabling a private marketplace experience

When a private marketplace is enabled (and has a procurement policy), users in associated accounts can only purchase products that you have approved. When no private marketplace experience is enabled for an account, users can purchase products across the full AWS Marketplace catalog.

To enable a private marketplace, use the `StartChangeRequest` action with the `UpdateExperience` change type.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json
```

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateExperience",
      "DetailsDocument":
      {
        "Status": "Enabled"
      },
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@4"
      }
    }
  ]
}
```

Similarly, you can use the same action and `ChangeType`, but change the `Status` in `DetailsDocument` to `Disabled` to disable a private marketplace.

Note

Disabling a private marketplace keeps your list of both allowed and denied products, as well as customizations, such as branding. When a private marketplace is disabled, users no longer see the private marketplace (although they may still be governed by the default experience for the organization). If there are no private marketplace experiences enabled for an account, then all restrictions are removed, and users are able to procure any products in the public AWS Marketplace.

Enabling or disabling user requests

Users in your organization can view the full public AWS Marketplace, but they can only subscribe to the products that you have allowed. By default, they can request that a product that is not in the private marketplace be added to it. These requests show up in the private marketplace administrator page ([Private Marketplace](#)), where you can decide whether to accept or deny the request (and whether to block further requests for the same product). You cannot see or respond to the requests by using the Catalog API.

You can enable or disable the ability for users to create requests for your private marketplace experience. Use `StartChangeSet` with the `UpdateProcurementPolicy` change type. The ability to make requests is disabled in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "UpdateProcurementPolicy",
      "DetailsDocument":
      {
        "Configuration":
        {
          "PolicyResourceRequests": "Deny"
        }
      },
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@5"
      }
    }
  ]
}
```

To enable the change request capability for users, use `Allow` instead of `Deny` in `PolicyResourceRequests`.

To learn how to get the current status of this setting, see the next section, [Getting a list of products in a private marketplace experience](#).

Getting a list of products in a private marketplace experience

The products allowed (and denied) in a private marketplace are part of the procurement policy in the `Experience` entity. To get the details about the procurement policies in a private marketplace, you first get the procurement policy identifier from the `Experience` entity, and then call `DescribeEntity` with that identifier.

To get the procurement policy identifier, use `DescribeEntity` on the Experience entity that you are interested in, as shown in the following command.

```
GET /DescribeEntity?catalog=AWSMarketplace&entityId=exp-example01
```

Following is an example response.

```
{
  "Details": "{\"Name\": \"New Private Marketplace\", \"Status\": \"Enabled\",
  \"ProcurementPolicies\": [\"procpolicy-123example456\"], \"BrandingSettings\":
  [\"brandsettings-456example123\"]}",
  "DetailsDocument":
  {
    "Name": "New Private Marketplace",
    "Status": "Enabled",
    "ProcurementPolicies":
    [
      "procpolicy-123example456"
    ],
    "BrandingSettings":
    [
      "brandsettings-456example123"
    ]
  },
  "EntityArn": "arn:<...>:AWSMarketplace/Experience/exp-example-01",
  "EntityIdentifier": "exp-example01@6",
  "EntityType": "Experience@1.0",
  "LastModifiedDate": "2021-01-13T20:31:36Z"
}
```

Note

The `DetailsDocument` attribute contains the entity details as a JSON object. The legacy `Details` attribute contains the same JSON object as a string.

You can use the returned `EntityId` for the procurement policy to get the details, as shown in the following command.

```
GET /DescribeEntity?catalog=AWSMarketplace&entityId=procpolicy-123example456
```

This returns the full details of the policy, including both allowed and denied products. Following is an example response.

```
{
  "Details": "{\\\"Name\\\":\\\"ExampleProcurementPolicy\\\", \\\"Statements\\\":[\\\"Effect
  \\\":\\\"Allow\\\",\\\"Resources\\\":[\\\"Type\\\":\\\"Product\\\",\\\"Ids\\\":[\\\"example1-1234-
  abcd-5678-90abcdef1234\\\"]],{\\\"Type\\\":\\\"Product\\\",\\\"Ids\\\":[\\\"example2-2345-
  bcde-6789-01bcdea2345\\\"]}]},{\\\"Effect\\\":\\\"Deny\\\",\\\"Resources\\\":[\\\"Type\\\":\\\"Product
  \\\",\\\"Ids\\\":[\\\"example3-3456-cdef-7890-12defabc5678\\\"]}]},{\\\"Configuration\\\":
  {\\\"PolicyResourceRequests\\\":\\\"Allow\\\"}}\",
  "DetailsDocument":
  {
    "Name": "ExampleProcurementPolicy",
    "Statements":
    [
      {
        "Effect": "Allow",
        "Resources":
        [
          {
            "Type": "Product",
            "Ids":
            [
              "example1-1234-abcd-5678-90abcdef1234"
            ]
          },
          {
            "Type": "Product",
            "Ids":
            [
              "example2-2345-bcde-6789-01bcdea2345"
            ]
          }
        ]
      },
      {
        "Effect": "Deny",
        "Resources":
        [
          {
            "Type": "Product",
            "Ids":
            [

```



```

        "example3-3456-cdef-7890-12defabc5678"
      ]
    }
  ]
}
],
"Configuration":
{
  "PolicyResourceRequests": "Allow"
}
},
"EntityArn": "arn:<...>AWSMarketplace/ProcurementPolicy/procpolicy-123example456",
"EntityIdentifier": "procpolicy-123example456@4",
"EntityType": "ProcurementPolicy@1.0",
"LastModifiedDate": "2020-10-01T12:00:00Z"
}

```

In this example, the procurement policy has two allowed products and one denied product. The policy allows user resource requests.

Adding or removing products from a private marketplace

By default, a private marketplace does not have any approved products in it. Use change requests to add or remove a product. To add a product, use the `AllowProductProcurement` change type. To remove a product, use the `DenyProductProcurement` change type.

The following code example shows the `AllowProductProcurement` change type with the `StartChangeSet` action to add a product to a private marketplace.

```

POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "AllowProductProcurement",
      "DetailsDocument":
      {
        "Products":
        [

```

```
{
  "Ids":
  [
    "example-1234-abcd-5678-90abcded1234"
  ],
  "Notes": "Useful product"
}
],
"Entity":
{
  "Identifier": "exp-1234example@6",
  "Type": "Experience@1.0"
}
]
}
```

You add the product to the Experience entity for a private marketplace by using `AllowProductProcurement`. The syntax to remove a product from a private marketplace is identical, with the exception that you use the `DenyProductProcurement ChangeType` instead of `AllowProductProcurement`. The products are added to the allow (or deny) list of the `ProcurementPolicy` entity that is contained by your Experience entity.

Note

The list of products in the `DetailsDocument` of your change is an array of `Ids`, so you can add (or remove) multiple products with one call by including a list of product identifiers. The limit is 50 products in a single request.

The `Notes` field for the list of `Ids` is not required. However, you can use it to record why a decision to allow or deny a set of products was made.

Finding products

By getting the details of your procurement policy, you can find the product IDs for the products that are already in a private marketplace. However, the AWS Marketplace Catalog API does not provide a way to find the product IDs for other products. There are two ways to get product IDs to use with the Catalog API service:

- **Public marketplace** – After you find a product in the public marketplace, choose **Continue to Subscribe** to see a details page about the product (it will not subscribe you to the product). The URL will include the product ID as a parameter. For example, in the URL `https://aws.amazon.com/marketplace/fulfillment?productId=ab1234cd-1234-abcd-5678-90abcdef1234&ref_=dtl_psb_continue`, *ab1234cd-1234-abcd-5678-90abcdef1234* is the product ID.
- **AWS Marketplace Discovery API** – Programmatically, you can access the full list of products in the AWS Marketplace by using the Discovery API. The Discovery API is a private API. You must request access to be able to use it. For more information, see [Access control for the AWS Marketplace Discovery API](#).

Working with private marketplaces for AWS Organizations

Whether you are working with a private marketplace for your account or your organization, you use the same API. However, there are differences when working within your organization:

- Before you can use private marketplace feature in an organization, you must [enable trusted access](#) to provide private marketplace service (`private-marketplace.marketplace.amazonaws.com`) access to your AWS Organizations data. You must also [create the private marketplace service-linked role](#) in the management account. This role includes all the permissions that private marketplace requires to describe AWS Organizations and update private marketplace resources on your behalf. These actions can only be performed by the management account. It is recommended to perform this enablement using private marketplace administrator page. If you are a new customer, see [Private marketplaces](#) in the *AWS Marketplace Buyer Guide*. If you are an existing customer, see [Creating and managing a private marketplace](#) in the *AWS Marketplace Buyer Guide*.
- Private marketplace resources in an organization are created in the management account and shared with the member account that is a delegated administrator for private marketplace.
- When listing objects in a private marketplace from a member account that is a delegated administrator for private marketplace, you must specifically request them with the `SharedWithMe` filter. This applies to both `ListEntities` and `ListChangeSets` actions.

To list Experience objects in your own account, call `ListEntities` as shown in the following code example.

```
POST /ListEntities HTTP/1.1
```

```
Content-Type: application/json

{
  "Catalog": "AWSMarketplace",
  "EntityType": "Experience"
}
```

However, to list the entities that have been shared with you, you must add a `FilterList` with a `Scope` of `SharedWithMe`, as shown in the following code example. As a result, AWS Marketplace searches outside of your own account to find entities that are shared with you.

```
POST /ListEntities HTTP/1.1
Content-Type: application/json

{"Catalog": "AWSMarketplace",
 "EntityType": "Experience",
 "FilterList":
  [{
    "Name": "Scope",
    "ValueList":
      ["SharedWithMe"]
  ]}]
```

In this case, only entities outside of your account (the ones for your organization) are returned.

Similarly, to call `ListChangeSets`, you must set the scope, as shown in the following code example.

```
POST /ListChangeSets HTTP/1.1
Content-Type: application/json

{"Catalog": "AWSMarketplace",
 "FilterList":
  [{
    "Name": "Scope",
    "ValueList":
      ["SharedWithMe"]
  ]}]
```

This returns change sets that apply to a shared private marketplace for your organization.

Associating principals to experiences

A private marketplace experience must have one or more principals associated with it in order to have any effects in your organization. For a single AWS account, you must associate the account with the experience to use the private marketplace. In an organization, you can have multiple experiences apply to different principals.

Note

- The experience that is associated with the organization is the default for all other accounts in the organization. Associating a member account or OU with a different experience directly sets a different experience for the member account or child accounts of the OU.
- If you are a current private marketplace customer without the AWS Organizations integration for private marketplace, the experience that is associated with the management account is the default for all other accounts in the organization.

To associate a principal to an experience, use the `AssociateAudience` change type with the `StartChangeSet` action, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-example01@1"
      },
      "ChangeType": "AssociateAudience",
      "DetailsDocument":
      {
        "Name": "AudienceName",
        "Description": "Audience example.",
        "Principals":
```

```
    [
      "012345678901",
      "ou-abcd-01234567",
      "o-0123456789"
    ]
  }
},
"ChangeSetName": "Set Audience for experience 01"
}
```

The *audience* is the list of *principals* that are associated with the Experience. A principal is an AWS account, organizational unit, or organization defined by its ID. *Principals* is a list, so you can include multiple principals to be associated with the experience. After the first call, subsequent calls to the `AssociateAudience` change type will add principals to the association for the experience.

You can also remove accounts from an experience. Use the `DisassociateAudience` change type to do this, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-example01@02"
      },
      "ChangeType": "DisassociateAudience",
      "DetailsDocument":
      {
        "Principals":
        [
          "012345678901",
          "ou-abcd-01234567",
          "o-0123456789"
        ]
      }
    }
  ]
}
```

```
    }
  }
],
"ChangeSetName": "Disassociate audience example"
}
```

Note

A principal can only be directly associated with one experience. To move a principal from being directly associated with one experience to another experience, you must disassociate it from the initial experience, then associate it with the second.

Archiving and reactivating a private marketplace experience

You can remove a private marketplace experience by archiving it. Archived experiences can't be updated or used to govern accounts in your organization. If you have audiences associated with an archived experience, you can associate them with a different experience. If you decide to use the experience at a later time, you can always reactivate it. Administrators from the management account or a member account that is a delegated administrator for private marketplace have permissions to archive and reactivate experiences. If you're a current private marketplace customer without the AWS Organizations integration for private marketplace, administrators from the account that created the experience have permissions to archive and reactivate experiences.

Note

Before archiving an experience, you must disable it. For information about disabling an experience, see [Configuring your private marketplace](#) in the *AWS Marketplace Buyer Guide*.

To archive an experience, use the `RestrictExperience` change type with the `StartChangeSet` action, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
```

```
[
  {
    "ChangeType": "RestrictExperience",
    "DetailsDocument":
    {},
    "Entity":
    {
      "Type": "Experience@1.0",
      "Identifier": "exp-1234example"
    }
  }
]
```

To reactivate an experience, use the `ReviveExperience` change type with the `StartChangeSet` action, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "ReviveExperience",
      "DetailsDocument":
      {},
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example"
      }
    }
  ]
}
```

Errors in the private marketplace API

The following errors are specific to the private marketplace actions in the AWS Marketplace Catalog API.

Change type	Error code	Error message	Description
-------------	------------	---------------	-------------

Errors returned directly by the StartChangeSet action

All	422	Document not valid JSON format	Invalid JSON input used, check your syntax.
AllowProductProcurement, DenyProductProcurement	422	Values in Ids array must be unique	You can't include the same product multiple times in a single change request.
AllowProductProcurement, DenyProductProcurement	422	Cumulative number of values in Ids array must be less than or equal to 50	You can allow or deny up to 50 products in a single change request.

Errors found by calling the DescribeChangeSet action

CreateBrandingSettings, UpdateBrandingSettings	INVALID_URL	Image could not be fetched from the input URL	You must specify a valid, reachable URL for the logo field in BrandingSettings .
CreateBrandingSettings, UpdateBrandingSettings	INVALID_IMAGE	Image verification for type, content, or file size failed. Only .png and .svg file types with sizes less than or equal to 500KB are supported.	Your image file must match the logo requirements for branding settings.
AllowProductProcurement, DenyProductProcurement	ENTITY_NOT_FOUND	Procurement policy missing from Experience	You must create a ProcurementPolicy before

Change type	Error code	Error message	Description
			allowing or denying products.
CreateProcurementPolicy	ENTITY_ALREADY_EXISTS	Procurement policy exists for Experience	You can only have a single procurement policy for a private marketplace.
UpdateProcurementPolicy	ENTITY_NOT_FOUND	Procurement policy missing from Experience	You must create a ProcurementPolicy before updating the procurement policy.
CreateBrandingSettings	ENTITY_ALREADY_EXISTS	Branding settings exists for Experience	You can only have a single branding settings for a private marketplace.
UpdateBrandingSettings	ENTITY_NOT_FOUND	Branding settings missing from Experience	You must create a BrandingSettings entity before updating the branding settings.
AssociateAudience	CALLER_NOT_AUTHORIZED	Caller not authorized to execute the action	You must have permissions to call the action. The accounts being added must be in the same organization.
CreateExperience	CALLER_NOT_AUTHORIZED	Caller not authorized to create experience.	You must have permissions to create an experience.

Change type	Error code	Error message	Description
AssociateAudience	ENTITY_ALREADY_EXISTS	An experience is already associated with the account {accountId}. Disassociate previous experience before updating	You can only associate a single experience with an account. Disassociate the current experience before associating a new one.
AssociateAudience, DisassociateAudience	ENTITY_IN_USE	There is already a conflicting change in progress for the selected account. Try again later	You can't change the association with an account while another change request to change the association is already in progress.

Entity types defined by private marketplace

The following table lists the private marketplace entity types, purpose, and actions on which each can be specified. Each entity type can be used to specify a resource Amazon Resource Name (ARN) that can be used in the AWS Identity and Access Management (IAM) policy. For more details on ARN formats, see [the section called "Catalog API entities"](#).

Entity	Purpose	Actions
Experience	Stores the top-level settings for a private marketplace	StartChangeSet DescribeEntity
BrandingSettings	Stores the branding settings for a private marketplace	DescribeEntity
ProcurementPolicy	Stores the procurement settings and lists of products in a private marketplace	DescribeEntity

Entity	Purpose	Actions
Audience	Stores the details of principals associated with a private marketplace	DescribeEntity

Using the AWS Marketplace API to share resources

AWS Marketplace Catalog API integrates with AWS Resource Access Manager (AWS RAM) to enable resource sharing. A *resource* is an entity that users can work with in AWS Marketplace, such as a product, an offer, or a resale authorization. With AWS RAM, you can share some AWS Marketplace Catalog API resources with other AWS accounts. You share resources that you own by creating a *resource share*. A resource share specifies the resources that you want to share and the consumers with whom to share them.

Contents

- [Prerequisites to share AWS Marketplace entities](#)
- [Share an AWS Marketplace entity](#)

Prerequisites to share AWS Marketplace entities

Before sharing entities in AWS Marketplace Catalog API, you must meet the following prerequisites:

- You can only have one resource policy attached to your AWS Marketplace entity.
- To share an AWS Marketplace entity, you must own it in your AWS account. This requirement means that the entity must be allocated or provisioned in your account. You can't share an AWS Marketplace entity that has been shared with you.

Share an AWS Marketplace entity

With AWS Marketplace resource sharing, entity owners can share their entities with other AWS accounts in AWS Marketplace. Entity-owners can be ISVs and channel partners. Entities that can be shared are products, offers, and resale authorizations.

Note

At this time, you can only share entities. Entities in AWS Marketplace include `AmiProduct`, `Audience`, `BrandingSettings`, `ContainerProduct`, `Experience`, and `ProcurementPolicy`.

For more information about AWS RAM, see the [AWS RAM User Guide](#). For more information about managing your shared resources, see [Using shared AWS resources](#) in the *AWS RAM User Guide*.

As a *sharing account*, you can set read-only or both read/write on the resources that you want to share. These permissions determine what operations a *consuming account* can perform on the resources that are shared with them.

- **Sharing account** – The resource that is shared and in which the AWS RAM administrator creates the AWS resource share by using AWS RAM.
- **Consuming account** – The AWS account to which a resource is shared. The resource share can specify an entire account as the principal, or for some resource types, individual roles or users in the account.

To share an AWS Marketplace entity, you must add it to a resource share. A resource share is an AWS RAM resource that lets you share your resources across AWS accounts. A resource share specifies the resources to share, and the consumers with whom they are shared. When you share an entity using the AWS Marketplace console, you add it to an existing resource share. To add the AWS Marketplace entity to a new resource share, you must first create the resource share using the [AWS RAM console](#).

You can share an AWS Marketplace entity that you own using the AWS Marketplace console, AWS RAM console, or the AWS Command Line Interface (AWS CLI).

To share an AWS Marketplace entity that you own using the AWS RAM console

See [Creating a Resource Share](#) in the *AWS RAM User Guide*.

To share an AWS Marketplace entity that you own using the AWS CLI

Use the [create-resource-share](#) command.

Note

For resource types such as entities that support resource-based policies, you can use AWS RAM to share resources to use additional AWS RAM features. For more information, see [Resource-based policy](#) in the *AWS RAM User Guide*. AWS RAM uses the AWS Marketplace Catalog API to automatically construct the resource policy from permissions in a resource share and manages that resource policy for you.

For information about how to set, view, or delete AWS resource-based policies on your AWS Marketplace entity through AWS RAM, see [Allowing actions on all resources](#) in the *AWS RAM User Guide*.

Differences between sharing an entity through AWS RAM and the AWS Marketplace Catalog API

In addition to sharing your entity through AWS RAM, you can also set, view, or delete AWS resource-based policies on your entities through the AWS Marketplace Catalog API. However, there are a few differences between sharing your entity through AWS RAM and through the AWS Marketplace Catalog API.

When you share an entity through AWS RAM:

- If you share your entity with accounts that are outside of AWS Organizations, the consuming account must first accept your sharing request before the entity is shared.
- The consuming account can discover the shared entity through `ListEntities` with `OwnershipType` set to `SHARED`.
- You must adhere to several resource quotas. For more information, see [Service quotas for AWS RAM](#) in the *AWS RAM User Guide*.

When you share an entity through the AWS Marketplace Catalog API:

- Your entity will be shared as soon as the `PutResourcePolicy` request succeeds with no input from the consuming account.
- The consuming account can't discover the shared entity through `ListEntities` with `OwnershipType` set to `SHARED`. Instead, the owner of the sharing account must inform the consuming account of the shared entity ID.

Note

If your use case requires sharing resources that might exceed AWS RAM service quotas, or if you want to share resources without direct input from the consuming account, consider sharing through the AWS Marketplace Catalog API. For all other use cases, consider using AWS RAM to share AWS Marketplace resources.

The following sections detail how you can set, view, or delete AWS resource-based policies on your entities through the AWS Marketplace Catalog API.

Topics

- [Attach read-only policy to your resource](#)
- [Attach read and write resource policy to your resource](#)
- [View resource policy set on your resource](#)
- [Delete resource policy on your resource](#)
- [View all resources owned by you and shared with you](#)

Attach read-only policy to your resource

You can create a read-only resource-based policy on your shared resource using a sharing account. With this policy, the principal can only view the details of the resource that is shared with them.

Request

```
POST /PutResourcePolicy HTTP/1.1
Content-type: application/json

{
  "ResourceArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
AmiProduct/example2-abcd-1234-5ef6",
  "Policy": {
    "Version": "2012-10-17",
    "Statement": {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::222233334444:root"
      },
      "Action": [
        "aws-marketplace:DescribeEntity"
      ],
      "Resource": [
        "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
AmiProduct/example2-abcd-1234-5ef6"
      ]
    }
  }
}
```



```
}
```

Response

```
HTTP/1.1 200  
Content-type: application/json
```

```
{}
```

Attach read and write resource policy to your resource

As a sharing account, you can create a read and write resource-based policy on your shared resource. With this policy, the principal can view the details and perform write operations on the resource that is shared with them.

Request

```
POST /PutResourcePolicy HTTP/1.1  
Content-type: application/json  
  
{  
  "ResourceArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/  
AmiProduct/example2-abcd-1234-5ef6",  
  "Policy": {  
    "Version": "2012-10-17",  
    "Statement": {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::222233334444:root"  
      },  
      "Action": [  
        "aws-marketplace:DescribeEntity",  
        "aws-marketplace:StartChangeSet"  
      ],  
      "Resource": [  
        "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/  
AmiProduct/example2-abcd-1234-5ef6"  
      ]  
    }  
  }  
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{}
```

View resource policy set on your resource

As a sharing account, you can view the resource policy that is set on your shared resource.

Request

```
POST /GetResourcePolicy HTTP/1.1
Content-type: application/json

{
  "ResourceArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
  AmiProduct/example2-abcd-1234-5ef6"
}
```

Response

```
HTTP/1.1 200
Content-type: application/json

{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::222233334444:root"
      },
      "Action": [
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Resource": [
        "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
        AmiProduct/example2-abcd-1234-5ef6"
      ]
    }
  }
}
```

```
    }  
  }  
}
```

Delete resource policy on your resource

As a sharing account, you can delete the resource policy that is set on your shared resource.

Request

```
POST /DeleteResourcePolicy HTTP/1.1  
Content-type: application/json  
  
{  
  "ResourceArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/  
AmiProduct/example2-abcd-1234-5ef6"  
}
```

Response

```
HTTP/1.1 200  
Content-type: application/json  
  
{}
```

View all resources owned by you and shared with you

As a consuming account, you can view the resources that are shared with you.

Note

You can view the resources shared with you only if the resources were shared through AWS RAM.

Request

```
POST /ListEntities HTTP/1.1
```

```
Content-type: application/json
{
  "Catalog": "AWSMarketplace",
  "EntityType": "AmiProduct",
  "FilterList": [
    {
      "Name": "EntityId",
      "ValueList": [ "example2-abcd-1234-5ef6" ]
    }
  ],
  "OwnershipType": "SHARED"
}
```

Response


```
HTTP/1.1 200
Content-type: application/json

{
  "EntitySummaryList": [
    {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
AmiProduct/example2-abcd-1234-5ef6",
      "EntityId": "example2-abcd-1234-5ef6",
      "EntityType": "AmiProduct",
      "LastModifiedDate": "2018-02-27T13:45:22Z",
      "Name": "TestProduct",
      "Visibility": "public"
    }
  ],
  "NextToken": ""
}
```

Using this service with an AWS SDK

AWS software development kits (SDKs) are available for many popular programming languages. Each SDK provides an API, code examples, and documentation that make it easier for developers to build applications in their preferred language.

SDK documentation	Code examples
AWS SDK for C++	AWS SDK for C++ code examples
AWS CLI	AWS CLI code examples
AWS SDK for Go	AWS SDK for Go code examples
AWS SDK for Java	AWS SDK for Java code examples
AWS SDK for JavaScript	AWS SDK for JavaScript code examples
AWS SDK for Kotlin	AWS SDK for Kotlin code examples
AWS SDK for .NET	AWS SDK for .NET code examples
AWS SDK for PHP	AWS SDK for PHP code examples
AWS Tools for PowerShell	Tools for PowerShell code examples
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) code examples
AWS SDK for Ruby	AWS SDK for Ruby code examples
AWS SDK for Rust	AWS SDK for Rust code examples
AWS SDK for SAP ABAP	AWS SDK for SAP ABAP code examples
AWS SDK for Swift	AWS SDK for Swift code examples

 **Example availability**

Can't find what you need? Request a code example by using the **Provide feedback** link at the bottom of this page.

Code examples for AWS Marketplace using AWS SDKs

The following code examples show how to use AWS Marketplace with an AWS software development kit (SDK).

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Code examples

- [Code examples for AWS Marketplace Catalog API using AWS SDKs](#)
 - [AMI products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Add a dimension to an existing AMI product and update the offer pricing terms using an AWS SDK](#)
 - [Add a region where an AMI product is deployed using an AWS SDK](#)
 - [Create a public or limited AMI product and a public offer with hourly annual pricing using an AWS SDK](#)
 - [Create a public or limited AMI product and public offer with hourly monthly pricing using an AWS SDK](#)
 - [Create a public or limited AMI product and public offer with hourly pricing using an AWS SDK](#)
 - [Create an draft AMI product with a draft public offer using an AWS SDK](#)
 - [Restrict a region where an AMI product is deployed using an AWS SDK](#)
 - [Restrict product visibility using an AWS SDK](#)
 - [Specify whether AMI assets are deployed in new regions using an AWS SDK](#)
 - [Channel partner offers for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft CPPO for any product type using an AWS SDK](#)
 - [Create a resale authorization replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
 - [List all CPPOs created by a channel partner using an AWS SDK](#)
 - [List all shared resale authorizations available to a channel partner using an AWS SDK](#)
 - [Publish a CPPO and append a buyer EULA using an AWS SDK](#)

- [Publish a CPPO using one-time resale authorization and update price markup using an AWS SDK](#)
- [Publish a draft CPPO and update price markup using an AWS SDK](#)
- [Update the expiration date of a CPPO using an AWS SDK](#)
- [Container products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft container product with a draft public offer using an AWS SDK](#)
 - [Create a limited container product with public offer, contract pricing using an AWS SDK](#)
- [Entities for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Describe all entities in a single call using an AWS SDK](#)
 - [List and describe all offers associated with a product using an AWS SDK](#)
- [Offers for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a custom dimension for a SaaS product and create a private offer using an AWS SDK](#)
 - [Create a draft private offer for an AMI or SaaS product using an AWS SDK](#)
 - [Create a private offer with contract and Pay-As-You-Go pricing for a SaaS product using an AWS SDK](#)
 - [Create a private offer with contract pricing and a flexible payment schedule for a SaaS product using an AWS SDK](#)
 - [Create a private offer with contract pricing for a Container product using an AWS SDK](#)
 - [Create a private offer with contract pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly annual pricing and a flexible payment schedule for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly annual pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with subscription pricing for a SaaS product using an AWS SDK](#)
 - [Create a private offer with tiered contract pricing for a SaaS product using an AWS SDK](#)
 - [Create a public free trial offer with subscription pricing for a SaaS product using an AWS SDK](#)
 - [Create a replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
- [Describe a public offer using an AWS SDK](#)

- [Expire a private offer using an AWS SDK](#)
- [List all private offers using an AWS SDK](#)
- [List released public and private offers for a specific product ID using an AWS SDK](#)
- [Update an offer to apply a contract with Pay-As-You-Go pricing using an AWS SDK](#)
- [Update an offer to apply hourly annual pricing using an AWS SDK](#)
- [Update an offer to apply targeting to specific geographic regions using an AWS SDK](#)
- [Update name and description of a public offer using an AWS SDK](#)
- [Update the EULA of an offer using an AWS SDK](#)
- [Update the expiration date of a private offer to a future date using an AWS SDK](#)
- [Update the free trial duration of a public free trial offer for a SaaS product using an AWS SDK](#)
- [Update the refund policy of an offer using an AWS SDK](#)
- [Products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Describe an AMI, SaaS, or Container product using an AWS SDK](#)
 - [List all AMI, SaaS, or Container products and associated public offers using an AWS SDK](#)
- [Resale authorization for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create draft resale authorization for any product type using an AWS SDK](#)
 - [Describe a resale authorization using an AWS SDK](#)
 - [Publish a one-time resale authorization with a private offer using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date and a EULA to be sent to the buyer using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date and add reseller contract documentation using an AWS SDK](#)
 - [Publish multi-use resale authorization with expiration and add a specific buyer account for the resale using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date and add a custom EULA using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date and add reseller contract documentation using an AWS SDK](#)

- [Publish multi-use resale authorization without expiration date and add a specific buyer account for the resale using an AWS SDK](#)
- [Publish one-time resale authorization and add Flexible payment schedule using an AWS SDK](#)
- [Publish one-time resale authorization for any product type and add a EULA using an AWS SDK](#)
- [Publish one-time resale authorization and add a specific buyer account for the resale using an AWS SDK](#)
- [Publish one-time resale authorization for any product type and add reseller contract documentation using an AWS SDK](#)
- [Publish one-time resale authorization for and add whether it is a renewal using an AWS SDK](#)
- [Restrict resale authorization using an AWS SDK](#)
- [Update name and description of one-time or multi-use resale authorization using an AWS SDK](#)
- [SaaS products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft SaaS product with a draft public offer using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with contract pricing using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with contract with Pay-As-You-Go pricing using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with subscription pricing using an AWS SDK](#)
 - [Publish a SaaS product and associated public offer using an AWS SDK](#)
 - [Publish a SaaS product and associated public offer from an existing draft using an AWS SDK](#)
 - [Update dimensions on an AMI or SaaS product using an AWS SDK](#)
- [Utilities for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Utilities to start a changeset using an AWS SDK](#)
- [Code examples for AWS Marketplace Agreement API using AWS SDKs](#)
 - [Agreements for AWS Marketplace Agreement API using AWS SDKs](#)
 - [Get all agreement IDs using an AWS SDK](#)

- [Get all agreements using an AWS SDK](#)
- [Get customer ID from an agreement using an AWS SDK](#)
- [Get financial details from an agreement using an AWS SDK](#)
- [Get free trial details from an agreement using an AWS SDK](#)
- [Get information about an agreement using an AWS SDK](#)
- [Get product and offer details from an agreement using an AWS SDK](#)
- [Get the EULA of an agreement using an AWS SDK](#)
- [Get the auto renewal terms of an agreement using an AWS SDK](#)
- [Get the dimensions purchased in an agreement using an AWS SDK](#)
- [Get the instances of each dimension purchased in an agreement using an AWS SDK](#)
- [Get the payment schedule of an agreement using an AWS SDK](#)
- [Get the pricing per dimension in an agreement using an AWS SDK](#)
- [Get the pricing type of an agreement using an AWS SDK](#)
- [Get the product type of an agreement using an AWS SDK](#)
- [Get the status of an agreement using an AWS SDK](#)
- [Get the support terms of an agreement using an AWS SDK](#)
- [Get the terms of an agreement using an AWS SDK](#)
- [Search for agreements by account ID using an AWS SDK](#)
- [Search for agreements by agreement ID using an AWS SDK](#)
- [Search for agreements by end date using an AWS SDK](#)
- [Search for agreements by offer ID using an AWS SDK](#)
- [Search for agreements by product ID using an AWS SDK](#)
- [Search for agreements by status using an AWS SDK](#)
- [Search for agreements with one custom filter using an AWS SDK](#)
- [Search for agreements with two custom filters using an AWS SDK](#)

Code examples for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with an AWS software development kit (SDK).

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Code examples

- [AMI products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Add a dimension to an existing AMI product and update the offer pricing terms using an AWS SDK](#)
 - [Add a region where an AMI product is deployed using an AWS SDK](#)
 - [Create a public or limited AMI product and a public offer with hourly annual pricing using an AWS SDK](#)
 - [Create a public or limited AMI product and public offer with hourly monthly pricing using an AWS SDK](#)
 - [Create a public or limited AMI product and public offer with hourly pricing using an AWS SDK](#)
 - [Create a draft AMI product with a draft public offer using an AWS SDK](#)
 - [Restrict a region where an AMI product is deployed using an AWS SDK](#)
 - [Restrict product visibility using an AWS SDK](#)
 - [Specify whether AMI assets are deployed in new regions using an AWS SDK](#)
- [Channel partner offers for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft CPPO for any product type using an AWS SDK](#)
 - [Create a resale authorization replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
 - [List all CPPOs created by a channel partner using an AWS SDK](#)
 - [List all shared resale authorizations available to a channel partner using an AWS SDK](#)
 - [Publish a CPPO and append a buyer EULA using an AWS SDK](#)
 - [Publish a CPPO using one-time resale authorization and update price markup using an AWS SDK](#)
 - [Publish a draft CPPO and update price markup using an AWS SDK](#)
 - [Update the expiration date of a CPPO using an AWS SDK](#)
- [Container products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft container product with a draft public offer using an AWS SDK](#)
 - [Create a limited container product with public offer, contract pricing using an AWS SDK](#)

- [Entities for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Describe all entities in a single call using an AWS SDK](#)
 - [List and describe all offers associated with a product using an AWS SDK](#)
- [Offers for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a custom dimension for a SaaS product and create a private offer using an AWS SDK](#)
 - [Create a draft private offer for an AMI or SaaS product using an AWS SDK](#)
 - [Create a private offer with contract and Pay-As-You-Go pricing for a SaaS product using an AWS SDK](#)
 - [Create a private offer with contract pricing and a flexible payment schedule for a SaaS product using an AWS SDK](#)
 - [Create a private offer with contract pricing for a Container product using an AWS SDK](#)
 - [Create a private offer with contract pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly annual pricing and a flexible payment schedule for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly annual pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with hourly pricing for an AMI product using an AWS SDK](#)
 - [Create a private offer with subscription pricing for a SaaS product using an AWS SDK](#)
 - [Create a private offer with tiered contract pricing for a SaaS product using an AWS SDK](#)
 - [Create a public free trial offer with subscription pricing for a SaaS product using an AWS SDK](#)
 - [Create a replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
 - [Describe a public offer using an AWS SDK](#)
 - [Expire a private offer using an AWS SDK](#)
 - [List all private offers using an AWS SDK](#)
 - [List released public and private offers for a specific product ID using an AWS SDK](#)
 - [Update an offer to apply a contract with Pay-As-You-Go pricing using an AWS SDK](#)
 - [Update an offer to apply hourly annual pricing using an AWS SDK](#)
 - [Update an offer to apply targeting to specific geographic regions using an AWS SDK](#)
 - [Update name and description of a public offer using an AWS SDK](#)
 - [Update the FUI A of an offer using an AWS SDK](#)
- [Update the expiration date of a private offer to a future date using an AWS SDK](#)

- [Update the free trial duration of a public free trial offer for a SaaS product using an AWS SDK](#)
- [Update the refund policy of an offer using an AWS SDK](#)
- [Products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Describe an AMI, SaaS, or Container product using an AWS SDK](#)
 - [List all AMI, SaaS, or Container products and associated public offers using an AWS SDK](#)
- [Resale authorization for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create draft resale authorization for any product type using an AWS SDK](#)
 - [Describe a resale authorization using an AWS SDK](#)
 - [Publish a one-time resale authorization with a private offer using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date and a EULA to be sent to the buyer using an AWS SDK](#)
 - [Publish multi-use resale authorization with an expiration date and add reseller contract documentation using an AWS SDK](#)
 - [Publish multi-use resale authorization with expiration and add a specific buyer account for the resale using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date and add a custom EULA using an AWS SDK](#)
 - [Publish multi-use resale authorization without an expiration date and add reseller contract documentation using an AWS SDK](#)
 - [Publish multi-use resale authorization without expiration date and add a specific buyer account for the resale using an AWS SDK](#)
 - [Publish one-time resale authorization and add Flexible payment schedule using an AWS SDK](#)
 - [Publish one-time resale authorization for any product type and add a EULA using an AWS SDK](#)
 - [Publish one-time resale authorization and add a specific buyer account for the resale using an AWS SDK](#)
 - [Publish one-time resale authorization for any product type and add reseller contract documentation using an AWS SDK](#)
 - [Publish one-time resale authorization for and add whether it is a renewal using an AWS SDK](#)
 - [Restrict resale authorization using an AWS SDK](#)
 - [Update name and description of one time or multi use resale authorization using an AWS SDK](#)

- [SaaS products for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Create a draft SaaS product with a draft public offer using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with contract pricing using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with contract with Pay-As-You-Go pricing using an AWS SDK](#)
 - [Create a public or limited SaaS product and public offer with subscription pricing using an AWS SDK](#)
 - [Publish a SaaS product and associated public offer using an AWS SDK](#)
 - [Publish a SaaS product and associated public offer from an existing draft using an AWS SDK](#)
 - [Update dimensions on an AMI or SaaS product using an AWS SDK](#)
- [Utilities for AWS Marketplace Catalog API using AWS SDKs](#)
 - [Utilities to start a changeset using an AWS SDK](#)

AMI products for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Add a dimension to an existing AMI product and update the offer pricing terms using an AWS SDK](#)
- [Add a region where an AMI product is deployed using an AWS SDK](#)
- [Create a public or limited AMI product and a public offer with hourly annual pricing using an AWS SDK](#)
- [Create a public or limited AMI product and public offer with hourly monthly pricing using an AWS SDK](#)
- [Create a public or limited AMI product and public offer with hourly pricing using an AWS SDK](#)
- [Create an draft AMI product with a draft public offer using an AWS SDK](#)
- [Restrict a region where an AMI product is deployed using an AWS SDK](#)
- [Restrict product visibility using an AWS SDK](#)
- [Specify whether AMI assets are deployed in new regions using an AWS SDK](#)

Add a dimension to an existing AMI product and update the offer pricing terms using an AWS SDK

The following code examples show how to add a dimension to an existing AMI product and update the offer pricing terms.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDimensions",
      "Entity": {
        "Identifier": "prod-11111111111111",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": [
        {
          "Key": "m7g.8xlarge",
          "Description": "m7g.8xlarge",
          "Name": "m7g.8xlarge",
          "Types": [
            "Metered"
          ],
          "Unit": "Hrs"
        }
      ]
    },
    {
      "ChangeType": "UpdatePricingTerms",
```



```

"Entity": {
  "Type": "Offer@1.0",
  "Identifier": "offer-111111111111"
},
"DetailsDocument": {
  "PricingModel": "Usage",
  "Terms": [
    {
      "Type": "UsageBasedPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "RateCard": [
            {
              "DimensionKey": "m5.large",
              "Price": "0.15"
            },
            {
              "DimensionKey": "m7g.4xlarge",
              "Price": "0.45"
            },
            {
              "DimensionKey": "m7g.2xlarge",
              "Price": "0.45"
            },
            {
              "DimensionKey": "m7g.8xlarge",
              "Price": "0.55"
            }
          ]
        }
      ]
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDimensions",
      "Entity": {
        "Identifier": "prod-11111111111111",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": [
        {
          "Key": "m7g.8xlarge",
          "Description": "m7g.8xlarge",
          "Name": "m7g.8xlarge",
          "Types": [
            "Metered"
          ],
          "Unit": "Hrs"
        }
      ]
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
          {
            "Type": "UsageBasedPricingTerm",
```

```

        "CurrencyCode": "USD",
        "RateCards": [
            {
                "RateCard": [
                    {
                        "DimensionKey": "m5.large",
                        "Price": "0.15"
                    },
                    {
                        "DimensionKey": "m7g.4xlarge",
                        "Price": "0.45"
                    },
                    {
                        "DimensionKey": "m7g.2xlarge",
                        "Price": "0.45"
                    },
                    {
                        "DimensionKey": "m7g.8xlarge",
                        "Price": "0.55"
                    }
                ]
            }
        ]
    }
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to add a dimension to an existing
AMI product and update the offer pricing terms.
CAPI-23
"""

import os

```

```
import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Add dimension for AMI product")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Add a region where an AMI product is deployed using an AWS SDK

The following code examples show how to add a region where an AMI product is deployed.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddRegions",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": {
        "Regions": [
          "us-east-2",
          "us-west-2"
        ]
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddRegions",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      }
    }
  ]
}
```

```

        },
        "DetailsDocument": {
            "Regions": [
                "us-east-2",
                "us-west-2"
            ]
        }
    ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to add a region where my
AMI product is deployed
CAPI-25A
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,

```

```
        "Add a region where my AMI product is deployed",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited AMI product and a public offer with hourly annual pricing using an AWS SDK

The following code examples show how to create a public or limited AMI product and a public offer with hourly annual pricing. This example creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
```

```
    "Entity": {
      "Type": "AmiProduct@1.0"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Operating Systems"
      ],
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awsmp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "AddRegions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Regions": [
        "us-east-1"
      ]
    }
  },
  {
```



```

    "ChangeType": "AddInstanceTypes",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "InstanceTypes": [
        "t2.micro"
      ]
    }
  },
  {
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Version": {
        "VersionTitle": "Test AMI Version1.0",
        "ReleaseNotes": "Test AMI Version"
      },
      "DeliveryOptions": [
        {
          "Details": {
            "AmiDeliveryOptionDetails": {
              "AmiSource": {
                "AmiId": "ami-11111111111111111",
                "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
                "UserName": "ec2-user",
                "OperatingSystemName": "AMAZONLINUX",
                "OperatingSystemVersion": "10.0.14393",
                "ScanningPort": 22
              },
              "UsageInstructions": "Test AMI Version",
              "RecommendedInstanceType": "t2.micro",
              "SecurityGroups": [
                {
                  "IpProtocol": "tcp",
                  "IpRanges": [
                    "0.0.0.0/0"
                  ],
                  "FromPort": 10,

```

```

        "ToPort": 22
      }
    ]
  }
},
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": [
    {
      "Key": "t2.micro",
      "Description": "t2.micro",
      "Name": "t2.micro",
      "Types": [
        "Metered"
      ],
      "Unit": "Hrs"
    }
  ]
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111",
        "222222222222"
      ]
    }
  }
},
{
  "ChangeType": "ReleaseProduct",

```

```

    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with hourly-annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [

```

```

        {
            "DimensionKey": "t2.micro",
            "Price": "0.15"
        }
    ]
}
],
},
{
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
        {
            "Selector": {
                "Type": "Duration",
                "Value": "P365D"
            },
            "RateCard": [
                {
                    "DimensionKey": "t2.micro",
                    "Price": "150"
                }
            ],
            "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
            }
        }
    ]
}
],
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [

```

```

        {
            "Type": "StandardEula",
            "Version": "2022-07-14"
        }
    ]
}
},
{
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",
                "RefundPolicy": "Absolutely no refund, period."
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
          "Sample highlight"
        ],
        "SearchKeywords": [
          "Sample keyword"
        ],
        "Categories": [
          "Operating Systems"
        ],
        "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
        "VideoUrls": [
```

```
        "https://sample.amazonaws.com/awssmp-video-1"
    ],
    "AdditionalResources": []
  }
},
{
  "ChangeType": "AddRegions",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Regions": [
      "us-east-1"
    ]
  }
},
{
  "ChangeType": "AddInstanceTypes",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "InstanceTypes": [
      "t2.micro"
    ]
  }
},
{
  "ChangeType": "AddDeliveryOptions",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Version": {
      "VersionTitle": "Test AMI Version1.0",
      "ReleaseNotes": "Test AMI Version"
    },
    "DeliveryOptions": [
      {
        "Details": {
          "AmiDeliveryOptionDetails": {
```

```

        "AmiSource": {
            "AmiId": "ami-11111111111111111",
            "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
            "UserName": "ec2-user",
            "OperatingSystemName": "AMAZONLINUX",
            "OperatingSystemVersion": "10.0.14393",
            "ScanningPort": 22
        },
        "UsageInstructions": "Test AMI Version",
        "RecommendedInstanceType": "t2.micro",
        "SecurityGroups": [
            {
                "IpProtocol": "tcp",
                "IpRanges": [
                    "0.0.0.0/0"
                ],
                "FromPort": 10,
                "ToPort": 22
            }
        ]
    }
},
{
    "ChangeType": "AddDimensions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
        {
            "Key": "t2.micro",
            "Description": "t2.micro",
            "Name": "t2.micro",
            "Types": [
                "Metered"
            ],
            "Unit": "Hrs"
        }
    ]
}
]

```



```

    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111",
            "222222222222"
          ]
        }
      }
    },
    {
      "ChangeType": "ReleaseProduct",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",

```

```

        "Description": "Test public offer with hourly-annual pricing for
        AmiProduct using AWS Marketplace API Reference Code"
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Usage",
            "Terms": [
                {
                    "Type": "UsageBasedPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.micro",
                                    "Price": "0.15"
                                }
                            ]
                        }
                    ]
                },
                {
                    "Type": "ConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P365D"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.micro",
                                    "Price": "150"
                                }
                            ],
                            "Constraints": {
                                "MultipleDimensionSelection": "Allowed",

```

```

        "QuantityConfiguration": "Allowed"
    }
}
],
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",
                "RefundPolicy": "Absolutely no refund, period."
            }
        ]
    }
},
{

```

```

        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a public or limited AMI
product and public offer with hourly-annual pricing and standard or custom EULA
CAPI-06
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Create limited AMI product and public offer with hourly-annual pricing
and standard EULA",
    )

```

```
    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited AMI product and public offer with hourly monthly pricing using an AWS SDK

The following code examples show how to create a public or limited AMI product and public offer with hourly monthly pricing. This example creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      }
    },
  ],
}
```

```
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Operating Systems"
      ],
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awsmvp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "AddRegions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Regions": [
        "us-east-1"
      ]
    }
  },
  {
    "ChangeType": "AddInstanceTypes",
    "Entity": {
      "Type": "AmiProduct@1.0",
```

```

        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "InstanceTypes": [
            "t2.micro"
        ]
    }
},
{
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Version": {
            "VersionTitle": "Test AMI Version1.0",
            "ReleaseNotes": "Test AMI Version"
        },
        "DeliveryOptions": [
            {
                "Details": {
                    "AmiDeliveryOptionDetails": {
                        "AmiSource": {
                            "AmiId": "ami-111111111111111111",
                            "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
                            "UserName": "ec2-user",
                            "OperatingSystemName": "AMAZONLINUX",
                            "OperatingSystemVersion": "10.0.14393",
                            "ScanningPort": 22
                        },
                        "UsageInstructions": "Test AMI Version",
                        "RecommendedInstanceType": "t2.micro",
                        "SecurityGroups": [
                            {
                                "IpProtocol": "tcp",
                                "IpRanges": [
                                    "0.0.0.0/0"
                                ],
                                "FromPort": 10,
                                "ToPort": 22
                            }
                        ]
                    }
                }
            }
        ]
    }
}
]

```

```
    }
  }
}
}],
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": [
    {
      "Key": "t2.micro",
      "Description": "t2.micro",
      "Name": "t2.micro",
      "Types": [
        "Metered"
      ],
      "Unit": "Hrs"
    }
  ],
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111",
        "222222222222"
      ]
    }
  }
},
{
  "ChangeType": "ReleaseProduct",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
```



```

    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with hourly-monthly pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "t2.micro",
                  "Price": "0.15"
                }
              ]
            }
          ]
        }
      ]
    }
  }
}

```

```

        }
    ]
}
},
{
    "Type": "RecurringPaymentTerm",
    "CurrencyCode": "USD",
    "BillingPeriod": "Monthly",
    "Price": "15.0"
}
]
}
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
}
},
{
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",

```

```

        "RefundPolicy": "Absolutely no refund, period."
    }
}
],
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "CreateProduct",
            "ChangeName": "CreateProductChange",
            "Entity": {
                "Type": "AmiProduct@1.0"
            },
            "DetailsDocument": {}
        },
        {
            "ChangeType": "UpdateInformation",
            "Entity": {

```

```

        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
            "Sample highlight"
        ],
        "SearchKeywords": [
            "Sample keyword"
        ],
        "Categories": [
            "Operating Systems"
        ],
        "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
        "VideoUrls": [
            "https://sample.amazonaws.com/awsmvp-video-1"
        ],
        "AdditionalResources": []
    }
},
{
    "ChangeType": "AddRegions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Regions": [
            "us-east-1"
        ]
    }
},
{
    "ChangeType": "AddInstanceTypes",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "InstanceTypes": [
            "t2.micro"
        ]
    }
}

```

```

    ]
  }
},
{
  "ChangeType": "AddDeliveryOptions",
  "Entity": {
    "Type": "AmiProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Version": {
      "VersionTitle": "Test AMI Version1.0",
      "ReleaseNotes": "Test AMI Version"
    },
    "DeliveryOptions": [
      {
        "Details": {
          "AmiDeliveryOptionDetails": {
            "AmiSource": {
              "AmiId": "ami-111111111111111111",
              "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
              "UserName": "ec2-user",
              "OperatingSystemName": "AMAZONLINUX",
              "OperatingSystemVersion": "10.0.14393",
              "ScanningPort": 22
            },
            "UsageInstructions": "Test AMI Version",
            "RecommendedInstanceType": "t2.micro",
            "SecurityGroups": [
              {
                "IpProtocol": "tcp",
                "IpRanges": [
                  "0.0.0.0/0"
                ],
                "FromPort": 10,
                "ToPort": 22
              }
            ]
          }
        }
      }
    ]
  }
}

```

```
    },
    {
      "ChangeType": "AddDimensions",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": [
        {
          "Key": "t2.micro",
          "Description": "t2.micro",
          "Name": "t2.micro",
          "Types": [
            "Metered"
          ],
          "Unit": "Hrs"
        }
      ]
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111",
            "222222222222"
          ]
        }
      }
    },
    {
      "ChangeType": "ReleaseProduct",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "CreateOffer",
```

```

    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with hourly-monthly pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "t2.micro",
                  "Price": "0.15"
                }
              ]
            }
          ]
        }
      ]
    }
  },

```

```
        {
            "Type": "RecurringPaymentTerm",
            "CurrencyCode": "USD",
            "BillingPeriod": "Monthly",
            "Price": "15.0"
        }
    ]
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",
                "RefundPolicy": "Absolutely no refund, period."
            }
        ]
    }
},
```



```

    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a public or limited AMI
product and public offer with hourly-monthly pricing and standard or custom EULA
CAPI-08
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "create limited AMI product and public offer with hourly-monthly pricing
and standard EULA",
    )

if __name__ == "__main__":

```

```
main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited AMI product and public offer with hourly pricing using an AWS SDK

The following code examples show how to create a public or limited AMI product and public offer with hourly pricing. This example creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
```

```

    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Operating Systems"
      ],
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awsmvp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "AddRegions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Regions": [
        "us-east-1"
      ]
    }
  },
  {
    "ChangeType": "AddInstanceTypes",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "InstanceTypes": [

```

```

        "t2.micro"
    ]
}
},
{
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Version": {
            "VersionTitle": "Test AMI Version1.0",
            "ReleaseNotes": "Test AMI Version"
        },
        "DeliveryOptions": [
            {
                "Details": {
                    "AmiDeliveryOptionDetails": {
                        "AmiSource": {
                            "AmiId": "ami-111111111111111111",
                            "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
                            "UserName": "ec2-user",
                            "OperatingSystemName": "AMAZONLINUX",
                            "OperatingSystemVersion": "10.0.14393",
                            "ScanningPort": 22
                        },
                        "UsageInstructions": "Test AMI Version",
                        "RecommendedInstanceType": "t2.micro",
                        "SecurityGroups": [
                            {
                                "IpProtocol": "tcp",
                                "IpRanges": [
                                    "0.0.0.0/0"
                                ],
                                "FromPort": 10,
                                "ToPort": 22
                            }
                        ]
                    }
                }
            }
        ]
    }
}
]

```

```
    }
  },
  {
    "ChangeType": "AddDimensions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
      {
        "Key": "t2.micro",
        "Description": "t2.micro",
        "Name": "t2.micro",
        "Types": [
          "Metered"
        ],
        "Unit": "Hrs"
      }
    ]
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "ReleaseProduct",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
```

```

    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with hourly pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "t2.micro",
                  "Price": "0.15"
                }
              ]
            }
          ]
        }
      ]
    }
  }
]

```

```

        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "SupportTerm",
          "RefundPolicy": "Absolutely no refund, period."
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    }
  }
]

```

```

        },
        "DetailsDocument": {}
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
          "Sample highlight"
        ]
      }
    }
  ]
}

```



```
        "SearchKeywords": [
            "Sample keyword"
        ],
        "Categories": [
            "Operating Systems"
        ],
        "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
        "VideoUrls": [
            "https://sample.amazonaws.com/awssmp-video-1"
        ],
        "AdditionalResources": []
    }
},
{
    "ChangeType": "AddRegions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Regions": [
            "us-east-1"
        ]
    }
},
{
    "ChangeType": "AddInstanceTypes",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "InstanceTypes": [
            "t2.micro"
        ]
    }
},
{
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
```

```

    "Version": {
      "VersionTitle": "Test AMI Version1.0",
      "ReleaseNotes": "Test AMI Version"
    },
    "DeliveryOptions": [
      {
        "Details": {
          "AmiDeliveryOptionDetails": {
            "AmiSource": {
              "AmiId": "ami-11111111111111111",
              "AccessRoleArn":
"arn:aws:iam::111111111111:role/AWSMarketplaceAmiIngestion",
              "UserName": "ec2-user",
              "OperatingSystemName": "AMAZONLINUX",
              "OperatingSystemVersion": "10.0.14393",
              "ScanningPort": 22
            },
            "UsageInstructions": "Test AMI Version",
            "RecommendedInstanceType": "t2.micro",
            "SecurityGroups": [
              {
                "IpProtocol": "tcp",
                "IpRanges": [
                  "0.0.0.0/0"
                ],
                "FromPort": 10,
                "ToPort": 22
              }
            ]
          }
        }
      }
    ]
  },
  {
    "ChangeType": "AddDimensions",
    "Entity": {
      "Type": "AmiProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
      {
        "Key": "t2.micro",

```

```

        "Description": "t2.micro",
        "Name": "t2.micro",
        "Types": [
            "Metered"
        ],
        "Unit": "Hrs"
    }
]
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "111111111111",
                "222222222222"
            ]
        }
    }
},
{
    "ChangeType": "ReleaseProduct",
    "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
},
{
    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
        "Type": "Offer@1.0"
    },
    "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier"
    }
},
{
    "ChangeType": "UpdateInformation",

```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with hourly pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "t2.micro",
                  "Price": "0.15"
                }
              ]
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {

```

```

        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    },
    {
        "ChangeType": "UpdateSupportTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "Terms": [
                {
                    "Type": "SupportTerm",
                    "RefundPolicy": "Absolutely no refund, period."
                }
            ]
        }
    },
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
```

```
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to Create a public or limited AMI
product
and public offer with hourly pricing and standard or custom EULA
CAPI-07
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Create limited AMI product and public offer with hourly pricing and
standard EULA",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create an draft AMI product with a draft public offer using an AWS SDK

The following code examples show how to create an draft AMI product with a draft public offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product"
      }
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier",
        "Name": "Test Offer"
      }
    }
  ]
}
```

```
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier",
        "Name": "Test Offer"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.


```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create an AMI draft product
with a draft public offer.
CAPI-02
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "AMI draft product with draft public offer",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Restrict a region where an AMI product is deployed using an AWS SDK

The following code examples show how to restrict a region where an AMI product is deployed.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.


To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictRegions",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": {
        "Regions": [
          "us-west-2"
        ]
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictRegions",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": {
        "Regions": [
          "us-west-2"
        ]
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to restrict a region where my
AMI product is deployed
CAPI-25B
"""
```

```
import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Add a region where my AMI product is deployed",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Restrict product visibility using an AWS SDK

The following code examples show how to restrict product visibility.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateVisibility",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": {
        "TargetVisibility": "Restricted"
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateVisibility",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": {
        "TargetVisibility": "Restricted"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to change a product visibility to
restricted
CAPI-17
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usagem_demo(change_set, "Restrict existing AMI")
```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Specify whether AMI assets are deployed in new regions using an AWS SDK

The following code examples show how to specify whether AMI assets are deployed in new regions built by AWS to support future regions.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{  
  "Catalog": "AWSMarketplace",  
  "ChangeSet": [  
    {  
      "ChangeType": "UpdateFutureRegionSupport",  
      "Entity": {  
        "Type": "AmiProduct@1.0",  
        "Identifier": "prod-1111111111111111"  
      },  
      "DetailsDocument": {  
        "FutureRegionSupport": {  
          "SupportedRegions": [  
            "All"  
          ]  
        }  
      }  
    }  
  ]  
}
```

```

    ]
  }
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateFutureRegionSupport",
      "Entity": {
        "Type": "AmiProduct@1.0",
        "Identifier": "prod-111111111111"
      },
      "DetailsDocument": {
        "FutureRegionSupport": {
          "SupportedRegions": [
            "All"
          ]
        }
      }
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.


```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to modify a product to support
all future regions
CAPI-26
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Update future region support")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Channel partner offers for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Create a draft CPPO for any product type using an AWS SDK](#)

- [Create a resale authorization replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
- [List all CPPOs created by a channel partner using an AWS SDK](#)
- [List all shared resale authorizations available to a channel partner using an AWS SDK](#)
- [Publish a CPPO and append a buyer EULA using an AWS SDK](#)
- [Publish a CPPO using one-time resale authorization and update price markup using an AWS SDK](#)
- [Publish a draft CPPO and update price markup using an AWS SDK](#)
- [Update the expiration date of a CPPO using an AWS SDK](#)

Create a draft CPPO for any product type using an AWS SDK

The following code examples show how to create a draft CPPO for any product type so you can review them internally before publishing to buyers.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ResaleAuthorizationId": "11111111-1111-1111-1111-111111111111",

```

```

        "Name": "Test Offer",
        "Description": "Test product"
    }
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ResaleAuthorizationId": "11111111-1111-1111-1111-111111111111",
        "Name": "Test Offer name"
      }
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0

```

```
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create "draft" CPP0
for any product type (AMI/SaaS/Container) that can be reviewed internally
before publishing to buyers
CAPI-60
"""
import os

import utils.start_changeset as sc # noqa: E402
import utils.stringify_details as sd # noqa: E402

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usagem_demo(change_set, "Create a draft CPP0 offer for a product")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.


For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a resale authorization replacement private offer from an existing agreement with contract pricing using an AWS SDK

The following code examples show how to create a resale authorization replacement private offer from an existing agreement with contract pricing.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateReplacementOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateReplacementOfferResaleAuth",
      "DetailsDocument": {
        "AgreementId": "agmt-11111111111111111111111111111111",
        "ResaleAuthorizationId": "resaleauthz-1111111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test replacement offer for SaaSProduct using AWS
Marketplace API Reference Codes",
        "Description": "Test private resale replacement offer with
contract pricing for SaaSProduct"
      }
    },
    {

```

```

    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "FixedUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "Price": "0.0",
          "Duration": "P12M",
          "Grants": [
            {
              "DimensionKey": "BasicService",
              "MaxQuantity": 2
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ValidityTerm",
          "AgreementEndDate": "2024-01-30"
        }
      ]
    }
  },
  {
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
      "Type": "Offer@1.0",

```

```

        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "PaymentScheduleTerm",
                "CurrencyCode": "USD",
                "Schedule": [
                    {
                        "ChargeDate": "2024-01-01",
                        "ChargeAmount": "0"
                    }
                ]
            }
        ]
    },
    {
        "ChangeType": "UpdateLegalTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
        },
        "DetailsDocument": {
            "Terms": [
                {
                    "Type": "LegalTerm",
                    "Documents": [
                        {
                            "Type": "StandardEula",
                            "Version": "2022-07-14"
                        }
                    ]
                }
            ]
        }
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "Offer@1.0",

```

```

        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType" : "CreateReplacementOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateReplacementOfferResaleAuth",
      "DetailsDocument": {
        "AgreementId": "agmt-11111111111111111111111111111111",

```



```

        "ResaleAuthorizationId": "resaleauthz-111111111111"
    }
},
{
    "ChangeType": "UpdateInformation",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test replacement offer for SaaSProduct using AWS
Marketplace API Reference Codes",
        "Description": "Test private resale replacement offer with
contract pricing for SaaSProduct"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "FixedUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "Price": "0.0",
                "Duration": "P12M",
                "Grants": [
                    {
                        "DimensionKey": "BasicService",
                        "MaxQuantity": 2
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateValidityTerms",

```

```
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ValidityTerm",
          "AgreementEndDate": "2024-01-30"
        }
      ]
    }
  },
  {
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "PaymentScheduleTerm",
          "CurrencyCode": "USD",
          "Schedule": [
            {
              "ChargeDate": "2024-01-01",
              "ChargeAmount": "0"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
    },
    "DetailsDocument": {
```

```

        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-12-31"
        }
    },
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier":
"$CreateReplacementOfferResaleAuth.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose

```

Shows how to use the AWS SDK for Python (Boto3) to create a resale authorization replacement private offer from an existing agreement with contract pricing

CAPI-96

```
"""
```

```
import os
```

```
import utils.start_changeset as sc
```

```
import utils.stringify_details as sd
```

```
def main(change_set=None):
```

```
    if change_set is None:
```

```
        fname = "changeset.json"
```

```
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
```

```
        stringified_change_set = sd.stringify_changeset(change_set_file)
```

```
    else:
```

```
        stringified_change_set = change_set
```

```
    response = sc.usage_demo(
```

```
        stringified_change_set,
```

```
        "Create resale authorization replacement private offer with contract pricing",
```

```
    )
```

```
    return response
```

```
if __name__ == "__main__":
```

```
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List all CPPOs created by a channel partner using an AWS SDK

The following code examples show how to list all CPPOs created by a channel partner.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.core.document.Document;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.EntitySummary;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;

public class ListAllCppoOffers {

    /*
     * List all CPPOs created by a channel partner
     */
}
```

```
public static void main(String[] args) {

    List<String> cppoOfferIds = getAllCppoOfferIds();

    ReferenceCodesUtils.formatOutput(cppoOfferIds);
}

public static List<String> getAllCppoOfferIds() {
    MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    // get all offer entity ids
    List<String> entityIdList = new ArrayList<String>();

    ListEntitiesRequest listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER)
            .maxResults(10)
            .nextToken(null)
            .build();

    ListEntitiesResponse listEntitiesResponse =
        marketplaceCatalogClient.listEntities(listEntitiesRequest);

    for (EntitySummary entitySummary : listEntitiesResponse.entitySummaryList()) {
        entityIdList.add(entitySummary.entityId());
    }

    while (listEntitiesResponse.nextToken() != null) {
        listEntitiesRequest =
            ListEntitiesRequest.builder()
                .catalog(AWS_MP_CATALOG)
                .entityType(ENTITY_TYPE_OFFER)
                .maxResults(10)
                .nextToken(listEntitiesResponse.nextToken())
                .build();
        listEntitiesResponse =
            marketplaceCatalogClient.listEntities(listEntitiesRequest);

        for (EntitySummary entitySummary : listEntitiesResponse.entitySummaryList()) {
```

```
        entityIdList.add(entitySummary.entityId());
    }
}

// filter for CPP0 offers: ResaleAuthorizationId exists in Details

List<String> cppoOfferIds = new ArrayList<String>();

for (String entityId : entityIdList) {
    DescribeEntityRequest describeEntityRequest =
        DescribeEntityRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityId(entityId)
            .build();
    DescribeEntityResponse describeEntityResponse =
marketplaceCatalogClient.describeEntity(describeEntityRequest);

    Document resaleAuthorizationDocument =
describeEntityResponse.detailsDocument().asMap().get(ATTRIBUTE_RESALE_AUTHORIZATION_ID);
    String resaleAuthorizationId = resaleAuthorizationDocument != null ?
resaleAuthorizationDocument.asString() : "";

    if (!resaleAuthorizationId.isEmpty()) {
        cppoOfferIds.add(resaleAuthorizationId);
    }
}
return cppoOfferIds;
}
}
```

- For API details, see [ListEntities](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to list all Channel Partner
Offers
in an account

Program executed with no arguments:
ie. python3 list_all_cppo_offers.py

CAPI-93
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-catalog")

def get_offer_entities():
    """
    Returns a list of all offers in the account
    """

    next_token = "" # nosec: B105
    response_list = []

    try:
        response = mp_client.list_entities(Catalog="AWSMarketplace",
        EntityType="Offer")
    except ClientError as e:
        logging.exception(f"Couldn't list entities. {e}")
        raise

    response_list.append(response)

    # Results are paginated depending on number of entities returned
    while "NextToken" in response:
        next_token = response["NextToken"]
```



```
    try:
        response = mp_client.list_entities(
            Catalog="AWSMarketplace",
            EntityType="Offer",
            NextToken=next_token,
        )
    except ClientError as e:
        logging.exception(f"Couldn't list entities. {e}")
        raise

    if "NextToken" in response:
        response_list.append(response)

return response_list

def build_offer_list(response_list):
    """
    Cleans up list_entities response list with just list of offer IDs
    """
    offer_list = []

    for response in response_list:
        for entity in response["EntitySummaryList"]:
            offer_list.append(entity["EntityId"])

    return offer_list

def check_offer_resaleauth(offer_id):
    """
    Checks to see if an offer is based on a resale authorization
    """
    offer_response = describe_entity(offer_id)
    offer_details = json.loads(offer_response["Details"])
    if offer_details is None:
        offer_details = offer_response["DetailsDocument"]
    if "ResaleAuthorizationId" in offer_details and
offer_details["ResaleAuthorizationId"] is not None:
        return offer_id
    else:
        return None
```

```
def describe_entity(entity_id):
    """
    General purpose describe entity call
    """
    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace",
            EntityId=entity_id,
        )
    except ClientError as e:
        logging.exception(f"Couldn't describe entity. {e}")
        raise

    return response

def get_resaleauth_offers():
    """
    Returns a list of all offers in the account that are
    based on a resale authorization
    """
    resale_offer_list = []

    response_list = get_offer_entities()
    offer_list = build_offer_list(response_list)
    for offer in offer_list:
        print ("offer id " + offer)
        offer_info = check_offer_resaleauth(offer)
        if offer_info is not None:
            resale_offer_list.append(offer_info)

    return resale_offer_list

if __name__ == "__main__":
    print(get_resaleauth_offers())
```

- For API details, see [ListEntities](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List all shared resale authorizations available to a channel partner using an AWS SDK

The following code examples show how to list all shared resale authorizations available to a channel partner.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;

public class ListAllSharedResaleAuthorizations {

    /*
     * list all resale authorizations shared to an account
     */
}
```

```
public static void main(String[] args) {

    List<ListEntitiesResponse> responseList = getListEntityResponseList();
    ReferenceCodesUtils.formatOutput(responseList);
}

public static List<ListEntitiesResponse> getListEntityResponseList() {
    MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    List<ListEntitiesResponse> responseList = new
    ArrayList<ListEntitiesResponse>();

    ListEntitiesRequest listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_RESALE_AUTHORIZATION)
            .maxResults(10)
            .ownershipType(OWNERSHIP_TYPE_SHARED)
            .nextToken(null)
            .build();

    ListEntitiesResponse listEntitiesResponse =
    marketplaceCatalogClient.listEntities(listEntitiesRequest);

    responseList.add(listEntitiesResponse);

    while (listEntitiesResponse.nextToken() != null) {
        listEntitiesRequest = ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_RESALE_AUTHORIZATION)
            .maxResults(10)
            .ownershipType(OWNERSHIP_TYPE_SHARED)
            .nextToken(listEntitiesResponse.nextToken())
            .build();

        listEntitiesResponse =
        marketplaceCatalogClient.listEntities(listEntitiesRequest);

        responseList.add(listEntitiesResponse);
    }
}
```

```
    return responseList;
  }
}
```

- For API details, see [ListEntities](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to list all resale authorizations
shared to an account

Program executed with no arguments:
ie. python3 list_all_resale_authorizations.py

CAPI-94
"""

import logging

import boto3
import utils.helpers as hlp # noqa: E402
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-catalog")

def get_shared_entities():
    next_token = "" # nosec: B105
```

```
response_list = []

try:
    response = mp_client.list_entities(
        Catalog="AWSMarketplace",
        EntityType="ResaleAuthorization",
        OwnershipType="SHARED",
    )
except ClientError as e:
    logging.exception(f"Couldn't list entities. {e}")
    raise

response_list.append(response)

# Results can be paginated depending on number of entities returned
while "NextToken" in response:
    next_token = response["NextToken"]

    try:
        response = mp_client.list_entities(
            Catalog="AWSMarketplace",
            EntityType="ResaleAuthorization",
            OwnershipType="SHARED",
            NextToken=next_token,
        )
    except ClientError as e:
        logging.exception(f"Couldn't list entities. {e}")
        raise

    if "NextToken" in response:
        response_list.append(response)

return response_list

if __name__ == "__main__":
    response_list = get_shared_entities()
    hlp.pretty_print_datetime(response_list)
```

- For API details, see [ListEntities](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a CPPO and append a buyer EULA using an AWS SDK

The following code examples show how to publish a CPPO and append a buyer EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateCPP0offer",
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-111111111111",
        "Name": "Test Offer",
        "Description": "Test product"
      }
    },
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
      }
    }
  ]
}
```

```

    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/custom-eula.pdf"
            }
          ]
        }
      ]
    }
  ],
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": ["222222222222"]
    }
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-07-31"
  }
},
{
  "ChangeType": "UpdateValidityTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [

```



```

        {
            "Type": "ValidityTerm",
            "AgreementDuration": "P450D"
        }
    ]
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType" : "CreateOfferUsingResaleAuthorization",
            "Entity": {
                "Type": "Offer@1.0"
            },
            "ChangeName": "CreateCPP0offer",
            "DetailsDocument": {
                "ResaleAuthorizationId": "resaleauthz-111111111111",
                "Name": "Test Offer",
            }
        }
    ]
}

```

```
        "Description": "Test product"
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/custom-eula.pdf"
                    }
                ]
            }
        ]
    }
}
],
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": ["222222222222"]
        }
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-07-31"
    }
}
```

```

    },
    {
      "ChangeType": "UpdateValidityTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "ValidityTerm",
            "AgreementDuration": "P450D"
          }
        ]
      }
    },
    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to publish CPP0
for any product type (AMI/SaaS/Container) and append buyer EULA
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

```

```
def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Publish CPPO for any product type and append buyer EULA",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a CPPO using one-time resale authorization and update price markup using an AWS SDK

The following code examples show how to publish a CPPO using one-time resale authorization on AMI, SaaS, or Container products and update price markup.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType" : "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateCPP0offer",
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-11111111111111",
        "Name": "Test Offer",
        "Description": "Test product"
      }
    },
    {
      "ChangeType": "UpdateMarkup",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
      },
      "DetailsDocument": {
        "Percentage" : "5.0"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
```

```

        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": ["222222222222"]
        }
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-07-31"
    }
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementDuration": "P450D"
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateCPP0",
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test Offer name",
        "Description": "Test Offer description"
      }
    },
    {
      "ChangeType": "UpdateMarkup",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0.Entity.Identifier"
      },
    },
  ]
}
```

```

        "DetailsDocument": {
            "Percentage" : "5.0"
        }
    },
    {
        "ChangeType": "UpdateTargeting",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateCPP0.Entity.Identifier"
        },
        "DetailsDocument": {
            "PositiveTargeting": {
                "BuyerAccounts": ["111111111111"]
            }
        }
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateCPP0.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-12-31"
        }
    },
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateCPP0.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""

```


Purpose

Shows how to use the AWS SDK for Python (Boto3) to Create CPP0 using one-time resale

authorization on AMI, SaaS or Container products and update price markup

CAPI-63

```
"""
```

```
import os
```

```
import utils.start_changeset as sc
```

```
import utils.stringify_details as sd
```

```
def main(change_set=None):
```

```
    if change_set is None:
```

```
        fname = "changeset.json"
```

```
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
```

```
        stringified_change_set = sd.stringify_changeset(change_set_file)
```

```
    else:
```

```
        stringified_change_set = change_set
```

```
    response = sc.usage_demo(
```

```
        stringified_change_set,
```

```
        "Create CPP0 using one-time resale authorization and update price markup"
```

```
    )
```

```
    return response
```

```
if __name__ == "__main__":
```

```
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a draft CPPO and update price markup using an AWS SDK

The following code examples show how to publish a draft CPPO and update price markup.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType" : "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateCPP0offer",
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-1111111111111111",
        "Name": "Test Offer",
        "Description": "Test product"
      }
    },
    {
      "ChangeType": "UpdateMarkup",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0offer.Entity.Identifier"
      },
      "DetailsDocument": {
        "Percentage" : "5.0"
      }
    }
  ],
}
```

```

{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": ["222222222222"]
    }
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-07-31"
  }
},
{
  "ChangeType": "UpdateValidityTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "ValidityTerm",
        "AgreementDuration": "P450D"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateCPP0offer.Entity.Identifier"
  },
  "DetailsDocument": {}
}

```

```

    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOfferUsingResaleAuthorization",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateCPP0",
      "DetailsDocument": {
        "ResaleAuthorizationId": "resaleauthz-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateCPP0.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test Offer name",
        "Description": "Test Offer description"
      }
    },
    {
      "ChangeType": "UpdateMarkup",

```

```
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateCPP0.Entity.Identifier"
    },
    "DetailsDocument": {
      "Percentage": "5.0"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateCPP0.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": ["111111111111"]
      }
    }
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateCPP0.Entity.Identifier"
    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2023-05-31"
    }
  },
  {
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateCPP0.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ValidityTerm",
          "AgreementDuration": "P450D"
        }
      ]
    }
  }
}
```

```

    },
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateCPP0.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to publish "draft" CPP0
for any product type (AMI/SaaS/Container) and update price markup
CAPI-72
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Publish draft CPP0 for any product type adn update price markup"

```

```
)  
  
    return response  
  
if __name__ == "__main__":  
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update the expiration date of a CPPO using an AWS SDK

The following code examples show how to update the expiration date of a CPPO to give buyers more time to evaluate and accept the offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{  
  "Catalog": "AWSMarketplace",  
  "ChangeSet": [  
    {  
      "ChangeType": "UpdateAvailability",  
      "Entity": {  
        "Type": "Offer@1.0",
```

```

        "Identifier": "offer-111111111111"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2025-07-31"
    }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-111111111111"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2025-07-31"
      }
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
```



```
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to update the expiry
date of a CPP0 offer
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Update the expiry date of a CPP0 offer"
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Container products for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Create a draft container product with a draft public offer using an AWS SDK](#)
- [Create a limited container product with public offer, contract pricing using an AWS SDK](#)

Create a draft container product with a draft public offer using an AWS SDK

The following code examples show how to create a draft container product with a draft public offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "changeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product"
      }
    },
    {
```

```

        "ChangeType": "CreateOffer",
        "ChangeName": "CreateOfferChange",
        "Entity": {
            "Type": "Offer@1.0"
        },
        "DetailsDocument": {
            "ProductId": "$CreateProductChange.Entity.Identifier",
            "Name": "Test Offer"
        }
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
    }
  ]
}

```

```

        "DetailsDocument": {
            "ProductId": "$CreateProductChange.Entity.Identifier",
            "Name": "Test Offer"
        }
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create an container draft
product
with a draft public offer.
CAPI-03
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Create a draft container product with a draft public offer",
    )

    return response

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a limited container product with public offer, contract pricing using an AWS SDK

The following code examples show how to create a limited container product with a public offer, contract pricing, and standard EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {},
      "ChangeName": "CreateProductChange"
    },
  ],
}
```

```

{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
    "Categories": [
      "Streaming solutions"
    ],
    "ProductTitle": "ContainerProduct",
    "AdditionalResources": [],
    "LongDescription": "Long description goes here",
    "SearchKeywords": [
      "container streaming"
    ],
    "ShortDescription": "Description1",
    "Highlights": [
      "Highlight 1",
      "Highlight 2"
    ],
    "SupportDescription": "No support available",
    "VideoUrls": []
  }
},
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": [
    {
      "Key": "Cores",
      "Description": "Cores per cluster",
      "Name": "Cores",
      "Types": [
        "Entitled"
      ],
      "Unit": "Units"
    }
  ]
},

```

```
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111"
      ]
    }
  }
},
{
  "ChangeType": "AddRepositories",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Repositories": [
      {
        "RepositoryName": "uniquerepositoryname",
        "RepositoryType": "ECR"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseProduct",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {}
},
{
  "ChangeType": "CreateOffer",
  "Entity": {
    "Type": "Offer@1.0"
  },
  "DetailsDocument": {
    "ProductId": "$CreateProductChange.Entity.Identifier"
  }
}
```

```

    },
    "ChangeName": "CreateOfferChange"
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "Constraints": {
                "MultipleDimensionSelection": "Disallowed",
                "QuantityConfiguration": "Disallowed"
              },
              "RateCard": [
                {
                  "DimensionKey": "Cores",
                  "Price": "0.25"
                }
              ]
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {

```



```

        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    },
    {
        "ChangeType": "UpdateSupportTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "Terms": [
                {
                    "Type": "SupportTerm",
                    "RefundPolicy": "No refunds"
                }
            ]
        }
    },
    {
        "ChangeType": "UpdateInformation",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "Name": "Some container offer Name",
            "Description": "Some interesting container offer description"
        }
    },
    {
        "ChangeType": "UpdateRenewalTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        }
    }
}

```

```

    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "RenewalTerm"
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {},
    }
  ]
}

```

```

    "ChangeName": "CreateProductChange"
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "ContainerProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "Categories": [
        "Streaming solutions"
      ],
      "ProductTitle": "ContainerProduct",
      "AdditionalResources": [],
      "LongDescription": "Long description goes here",
      "SearchKeywords": [
        "container streaming"
      ],
      "ShortDescription": "Description1",
      "Highlights": [
        "Highlight 1",
        "Highlight 2"
      ],
      "SupportDescription": "No support available",
      "VideoUrls": []
    }
  },
  {
    "ChangeType": "AddDimensions",
    "Entity": {
      "Type": "ContainerProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
      {
        "Key": "Cores",
        "Description": "Cores per cluster",
        "Name": "Cores",
        "Types": [
          "Entitled"
        ],
        "Unit": "Units"
      }
    ]
  }
}

```

```
]
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111"
      ]
    }
  }
},
{
  "ChangeType": "AddRepositories",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Repositories": [
      {
        "RepositoryName": "uniquerepositoryname",
        "RepositoryType": "ECR"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseProduct",
  "Entity": {
    "Type": "ContainerProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {}
},
{
  "ChangeType": "CreateOffer",
  "Entity": {
    "Type": "Offer@1.0"
  },
}
```

```

    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    },
    "ChangeName": "CreateOfferChange"
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "Constraints": {
                "MultipleDimensionSelection": "Disallowed",
                "QuantityConfiguration": "Disallowed"
              },
              "RateCard": [
                {
                  "DimensionKey": "Cores",
                  "Price": "0.25"
                }
              ]
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    }
  }
}

```

```

    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "SupportTerm",
          "RefundPolicy": "No refunds"
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Some container offer Name",
      "Description": "Some interesting container offer description"
    }
  },
  {
    "ChangeType": "UpdateRenewalTerms",
    "Entity": {

```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "RenewalTerm"
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create limited container
product with public offer, contract pricing and standard EULA
CAPI-15
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):

```

```
if change_set is None:
    fname = "changeset.json"
    change_set_file = os.path.join(os.path.dirname(__file__), fname)
    stringified_change_set = sd.stringify_changeset(change_set_file)

else:
    stringified_change_set = change_set

response = sc.usage_demo(
    stringified_change_set,
    "Create limited container product with public offer contract pricing and
standard EULA",
)

return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Entities for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples


- [Describe all entities in a single call using an AWS SDK](#)
- [List and describe all offers associated with a product using an AWS SDK](#)

Describe all entities in a single call using an AWS SDK

The following code examples show how to describe all entities in a single call.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.catalogapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.BatchDescribeEntitiesRequest;
import software.amazon.awssdk.services.marketplacecatalog.model.EntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.BatchDescribeEntitiesResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.EntityDetail;
import
    software.amazon.awssdk.services.marketplacecatalog.model.BatchDescribeErrorDetail;

import java.util.Arrays;
import java.util.Map;

public class BatchDescribeEntities {

    /**
     * BatchDescribe my entities in a single call and
     * check if it contains all the information I need to know about the
     entities.
     */
    public static void main(String[] args) {
```

```
MarketplaceCatalogClient marketplaceCatalogClient =
    MarketplaceCatalogClient.builder()
        .httpClient(ApacheHttpClient.builder().build())
        .credentialsProvider(ProfileCredentialsProvider.create())
        .build();

BatchDescribeEntitiesRequest batchDescribeEntitiesRequest =
    BatchDescribeEntitiesRequest.builder()
        .entityRequestList(Arrays.asList(
            EntityRequest.builder()

.catalog(AWS_MP_CATALOG).entityId(OFFER_ID)
                .build(),
            EntityRequest.builder()

.catalog(AWS_MP_CATALOG).entityId(PRODUCT_ID)
                .build()))
        .build();

BatchDescribeEntitiesResponse batchDescribeEntitiesResponse =
marketplaceCatalogClient.batchDescribeEntities(batchDescribeEntitiesRequest);

// Reading the successful entities response
Map<String, EntityDetail> entityDetailsMap =
batchDescribeEntitiesResponse.entityDetails();
for (Map.Entry<String, EntityDetail> entry : entityDetailsMap.entrySet())
{
    System.out.println("EntityId: " + entry.getKey());
    ReferenceCodesUtils.formatOutput(entry.getValue());
}

// Logging the failed entities error details
Map<String, BatchDescribeErrorDetail> entityErrorsMap =
batchDescribeEntitiesResponse.errors();
for (Map.Entry<String, BatchDescribeErrorDetail> entry :
entityErrorsMap.entrySet()) {
    System.out.println(String.format("EntityId: %s, ErrorCode: %s,
ErrorMessage: %s", entry.getKey(),
        entry.getValue().errorCode(),
entry.getValue().errorMessage()));
}
}
```

- For API details, see [BatchDescribeEntities](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to describe for multiple entities
information in the AWS Marketplace Catalog
CAPI-98
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

PRODUCT_ID = "prod-111111111111"
OFFER_ID = "offer-111111111111"
MARKETPLACE_CATALOG = "AWSMarketplace"

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def get_entities_information(mp_client):
    """
    Returns information about a given product

```

```
Args: entity_id str: Entity to return
Returns: dict: Dictionary of product information
"""

entity_request_list_param = [
    {'EntityId': PRODUCT_ID, 'Catalog': MARKETPLACE_CATALOG},
    {'EntityId': OFFER_ID, 'Catalog': MARKETPLACE_CATALOG}
]
try:
    response = mp_client.batch_describe_entities(
        EntityRequestList=entity_request_list_param
    )

    return response

except ClientError as e:
    logger.exception("Unexpected error: %s", e)
    raise

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for entities in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")

    response = get_entities_information(mp_client)
    print("Successful entities response -")
    pretty_print(response["EntityDetails"])
    print("Failed entities response -")
    pretty_print(response["Errors"])

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [BatchDescribeEntities](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List and describe all offers associated with a product using an AWS SDK

The following code examples show how to list and describe all offers associated with a product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.EntitySummary;
import
    software.amazon.awssdk.services.marketplacecatalog.model.EntityTypeFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferFilters;
```

```
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferProductIdFilter;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferTargetingFilter;

public class ListProductPrivateOffers {

    private static MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();
    /*
     * retrieve all private offer information related to a single product
     */
    public static void main(String[] args) {

        List<EntitySummary> entitySummaryList = getEntitySummaryList();

        // for each offer id, output the offer detail using DescribeEntity API

        for (EntitySummary entitySummary : entitySummaryList) {
            DescribeEntityRequest describeEntityRequest =
                DescribeEntityRequest.builder()
                    .catalog(AWS_MP_CATALOG)
                    .entityId(entitySummary.entityId())
                    .build();
            DescribeEntityResponse describeEntityResponse =
                marketplaceCatalogClient.describeEntity(describeEntityRequest);
            ReferenceCodesUtils.formatOutput(describeEntityResponse);
        }
    }
    public static List<EntitySummary> getEntitySummaryList() {
        // define list entities filters

        EntityTypeFilters entityTypeFilters =
            EntityTypeFilters.builder()
                .offerFilters(OfferFilters.builder()
                    .targeting(OfferTargetingFilter.builder()
                        .valueListWithStrings(OFFER_TARGETING_BUYERACCOUNTS)
                        .build())
                    .productId(OfferProductIdFilter.builder()
                        .valueList(PRODUCT_ID)
                        .build())
                )
            )
    }
}
```

```
        .build())
    .build();

ListEntitiesRequest listEntitiesRequest =
    ListEntitiesRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .entityType(ENTITY_TYPE_OFFER).maxResults(50)
        .entityTypeFilters(entityTypeFilters)
        .nextToken(null)
        .build();

ListEntitiesResponse listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);

// save all entitySummary of the results into entitySummaryList

List<EntitySummary> entitySummaryList = new ArrayList<EntitySummary>();

entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());

while ( listEntitiesResponse.nextToken() != null &&
listEntitiesResponse.nextToken().length() > 0) {
    listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER).maxResults(50)
            .entityTypeFilters(entityTypeFilters)
            .nextToken(listEntitiesResponse.nextToken())
            .build();
    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
    entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());
}
return entitySummaryList;
}
}
```

- For API details, see the following topics in *AWS SDK for Java 2.x API Reference*.
 - [DescribeEntity](#)
 - [ListEntities](#)

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to retrieve all offer information
related to a single product
CAPI-97
"""

import argparse
import logging

import boto3
from botocore.exceptions import ClientError
from utils import helpers

logger = logging.getLogger(__name__)

mp_client = boto3.client("marketplace-catalog")

def get_entity_information(entity_id):
    """
    Returns information about a given entity
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of entity information
    """

    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace",
            EntityId=entity_id,
        )
```



```
        return response

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Entity with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def list_entity_details(entity_type, entity_id):
    """
    Returns details about a given entity and entity type
    """

    entity_summary_list = []

    # filter will return details for given entity_id with BuyerAccounts targeting
    filter_list_param = {
        'OfferFilters':{
            'ProductId':{
                'ValueList':[entity_id]
            },
            'Targeting': {
                'ValueList': ["BuyerAccounts"]
            }
        }
    }

    try:
        response = mp_client.list_entities(
            Catalog="AWSMarketplace",
            EntityType=entity_type,
            EntityTypeFilters = filter_list_param,
            MaxResults=10
        )

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Entity ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

    # add results to entity_summary_list
    entity_summary_list.extend(response["EntitySummaryList"])
```

```
# if there are more than 10 offers, paginate through the results
while "NextToken" in response and response["NextToken"] is not None:
    try:
        response = mp_client.list_entities(
            Catalog="AWSMarketplace",
            EntityType=entity_type,
            EntityTypeFilters = filter_list_param,
            NextToken=response["NextToken"],
            MaxResults=10
        )

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Entity ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

    # add results to entity_summary_list
    entity_summary_list.extend(response["EntitySummaryList"])

    return entity_summary_list

if __name__ == "__main__":
    parser = argparse.ArgumentParser()

    parser.add_argument(
        "--entity-id",
        "-eid",
        help="Provide Entity ID corresponding to a product to filter offers on",
        required=True,
    )

    args = parser.parse_args()

    # Gets a offers associated with the entity_id
    response = list_entity_details(
        "Offer",
        entity_id=args.entity_id
    )

    if response: # if response is not empty

        # list_entity_details returns a list of offers
```

```
for offer in response:

    print("-"*128)
    print(f"Terms for Offer ID: {offer['EntityId']}")
    print("-"*128)

    #retrieve offer information for each offer
    entity_information = get_entity_information(offer["EntityId"])

    helpers.pretty_print_datetime(entity_information)

else:
    print(f"No information found for Entity ID: {args.entity_id}")
```

- For API details, see the following topics in *AWS SDK for Python (Boto3) API Reference*.
 - [DescribeEntity](#)
 - [ListEntities](#)

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Offers for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Create a custom dimension for a SaaS product and create a private offer using an AWS SDK](#)
- [Create a draft private offer for an AMI or SaaS product using an AWS SDK](#)
- [Create a private offer with contract and Pay-As-You-Go pricing for a SaaS product using an AWS SDK](#)
- [Create a private offer with contract pricing and a flexible payment schedule for a SaaS product using an AWS SDK](#)
- [Create a private offer with contract pricing for a Container product using an AWS SDK](#)
- [Create a private offer with contract pricing for an AMI product using an AWS SDK](#)
- [Create a private offer with hourly annual pricing and a flexible payment schedule for an AMI product using an AWS SDK](#)

- [Create a private offer with hourly annual pricing for an AMI product using an AWS SDK](#)
- [Create a private offer with hourly pricing for an AMI product using an AWS SDK](#)
- [Create a private offer with subscription pricing for a SaaS product using an AWS SDK](#)
- [Create a private offer with tiered contract pricing for a SaaS product using an AWS SDK](#)
- [Create a public free trial offer with subscription pricing for a SaaS product using an AWS SDK](#)
- [Create a replacement private offer from an existing agreement with contract pricing using an AWS SDK](#)
- [Describe a public offer using an AWS SDK](#)
- [Expire a private offer using an AWS SDK](#)
- [List all private offers using an AWS SDK](#)
- [List released public and private offers for a specific product ID using an AWS SDK](#)
- [Update an offer to apply a contract with Pay-As-You-Go pricing using an AWS SDK](#)
- [Update an offer to apply hourly annual pricing using an AWS SDK](#)
- [Update an offer to apply targeting to specific geographic regions using an AWS SDK](#)
- [Update name and description of a public offer using an AWS SDK](#)
- [Update the EULA of an offer using an AWS SDK](#)
- [Update the expiration date of a private offer to a future date using an AWS SDK](#)
- [Update the free trial duration of a public free trial offer for a SaaS product using an AWS SDK](#)
- [Update the refund policy of an offer using an AWS SDK](#)

Create a custom dimension for a SaaS product and create a private offer using an AWS SDK

The following code examples show how to create a custom dimension for a SaaS product and create a private offer.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDimensions",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": [
        {
          "Types": [
            "Entitled"
          ],
          "Description": "Custom Pricing 4 w/ terms and coverage to be
defined in Private Offer",
          "Unit": "Units",
          "Key": "Custom4",
          "Name": "Custom Pricing 4"
        }
      ]
    },
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    }
  ]
}
```

```
    },
    "ChangeName": "CreateOfferChange"
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Private Test Offer - SaaS Contract Product",
      "Description": "Private Test Offer - SaaS Contract Product"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  }
}
```

```

    }
  ]
}
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Contract",
    "Terms": [
      {
        "Type": "ConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Constraints": {
              "MultipleDimensionSelection": "Allowed",
              "QuantityConfiguration": "Allowed"
            },
            "RateCard": [
              {
                "DimensionKey": "Custom4",
                "Price": "300.0"
              }
            ],
            "Selector": {
              "Type": "Duration",
              "Value": "P36M"
            }
          }
        ]
      }
    ]
  }
}
]

```

```

        }
      ]
    },
    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ],
  "ChangeSetName": "PrivateOfferWithCustomDimension"
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDimensions",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-11111111111111"
      },
      "DetailsDocument": [
        {
          "Types": [
            "Entitled"
          ],
        }
      ],
    }
  ],
}

```



```

        "Description": "Custom Pricing 4 w/ terms and coverage to be
defined in Private Offer",
        "Unit": "Units",
        "Key": "Custom4",
        "Name": "Custom Pricing 4"
    }
]
},
{
    "ChangeType": "CreateOffer",
    "Entity": {
        "Type": "Offer@1.0"
    },
    "DetailsDocument": {
        "ProductId": "prod-11111111111111"
    },
    "ChangeName": "CreateOfferChange"
},
{
    "ChangeType": "UpdateInformation",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Private Test Offer - SaaS Contract Product",
        "Description": "Private Test Offer - SaaS Contract Product"
    }
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "11111111111111"
            ]
        }
    }
},
{

```

```
"ChangeType": "UpdateLegalTerms",
"Entity": {
  "Type": "Offer@1.0",
  "Identifier": "$CreateOfferChange.Entity.Identifier"
},
"DetailsDocument": {
  "Terms": [
    {
      "Type": "LegalTerm",
      "Documents": [
        {
          "Type": "StandardEula",
          "Version": "2022-07-14"
        }
      ]
    }
  ]
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Contract",
    "Terms": [
      {
        "Type": "ConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Constraints": {
```

```

        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    },
    "RateCard": [
        {
            "DimensionKey": "Custom4",
            "Price": "300.0"
        }
    ],
    "Selector": {
        "Type": "Duration",
        "Value": "P36M"
    }
}
]
}
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
],
"ChangeSetName": "PrivateOfferWithCustomDimension"
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a SaaS product custom
dimension and private offer
CAPI-91
"""

```

```
import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "Create a SaaS product custom dimension and private offer"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a draft private offer for an AMI or SaaS product using an AWS SDK

The following code examples show how to create a draft private offer for an AMI or SaaS product so you can review it internally before publishing to buyers.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111",
        "Name": "Test Private Offer"
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111",

```

```

        "Name": "Test Private Offer"
    }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create "draft" Private Offer
for any AMI or SAAS product type that can be reviewed internally
before publishing to buyers
CAPI-30
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Private offer for AMI product")

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with contract and Pay-As-You-Go pricing for a SaaS product using an AWS SDK

The following code examples show how to create a private offer with contract and Pay-As-You-Go pricing for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      }
    }
  ]
}
```

```

    "DetailsDocument": {
      "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "WorkloadSmall",
                  "Price": "0.15"
                },
                {
                  "DimensionKey": "WorkloadMedium",
                  "Price": "0.25"
                }
              ]
            }
          ]
        }
      ]
    }
  }
]

```



```

        }
      ]
    },
  ],
  {
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P12M"
        },
        "RateCard": [
          {
            "DimensionKey": "BasicService",
            "Price": "150"
          },
          {
            "DimensionKey": "PremiumService",
            "Price": "300"
          }
        ],
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        }
      }
    ]
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",

```

```

        "Documents": [
            {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
        ]
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-12-31"
        }
    },
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111",
            "222222222222"
          ]
        }
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",

```

```
"Entity": {
  "Type": "Offer@1.0",
  "Identifier": "$CreateOfferChange.Entity.Identifier"
},
"DetailsDocument": {
  "PricingModel": "Contract",
  "Terms": [
    {
      "Type": "UsageBasedPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "RateCard": [
            {
              "DimensionKey": "WorkloadSmall",
              "Price": "0.15"
            },
            {
              "DimensionKey": "WorkloadMedium",
              "Price": "0.25"
            }
          ]
        }
      ]
    },
    {
      "Type": "ConfigurableUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P12M"
          },
          "RateCard": [
            {
              "DimensionKey": "BasicService",
              "Price": "150"
            },
            {
              "DimensionKey": "PremiumService",
              "Price": "300"
            }
          ]
        }
      ]
    }
  ]
}
```

```

        "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
        }
    }
]
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {

```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
contract with PAYG pricing for my SaaS product
CAPI-34
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Create private offer with contract with PAYG pricing for my SaaS
product",
    )

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with contract pricing and a flexible payment schedule for a SaaS product using an AWS SDK

The following code examples show how to create a private offer with contract pricing and a flexible payment schedule for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "111111111111",
                "222222222222"
            ]
        }
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "FixedUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "Price": "0.0",
                "Grants": [
                    {
                        "DimensionKey": "BasicService",
                        "MaxQuantity": 1
                    },
                    {

```



```

        "DimensionKey": "PremiumService",
        "MaxQuantity": 1
      }
    ]
  }
},
{
  "ChangeType": "UpdateValidityTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "ValidityTerm",
        "AgreementDuration": "P12M"
      }
    ]
  }
},
{
  "ChangeType": "UpdatePaymentScheduleTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",
        "Schedule": [
          {
            "ChargeDate": "2024-01-01",
            "ChargeAmount": "200.00"
          },
          {
            "ChargeDate": "2024-02-01",
            "ChargeAmount": "170.00"
          }
        ]
      }
    ]
  }
}

```

```

        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2023-12-31"
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]

```

```
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "FixedUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "Price": "0.0",
          "Grants": [
            {
              "DimensionKey": "BasicService",
              "MaxQuantity": 1
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {

```

```

        "Type": "ValidityTerm",
        "AgreementDuration": "P12M"
    }
]
}
},
{
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "PaymentScheduleTerm",
                "CurrencyCode": "USD",
                "Schedule": [
                    {
                        "ChargeDate": "2024-01-01",
                        "ChargeAmount": "200.00"
                    },
                    {
                        "ChargeDate": "2024-02-01",
                        "ChargeAmount": "170.00"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",

```

```

        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
    }
  ]
}
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "UpdateSupportTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "Some kind of refund policy description"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer
with contract pricing and flexible payment schedule for my SaaS product
CAPI-39
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Create private offer with contract pricing and flexible payment schedule
for my SaaS product",
    )

    return response

if __name__ == "__main__":
```

```
main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with contract pricing for a Container product using an AWS SDK

The following code examples show how to create a private offer with contract pricing for a Container product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
  ],
}
```



```

    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for Container product using AWS
Marketplace API Reference Code",
        "Description": "Test private offer for Container product with
contract pricing using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111"
          ]
        }
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
          {
            "Type": "ConfigurableUpfrontPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "Selector": {
                  "Type": "Duration",
                  "Value": "P12M"
                }
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```

    },
    "Constraints": {
      "MultipleDimensionSelection": "Disallowed",
      "QuantityConfiguration": "Disallowed"
    },
    "RateCard": [
      {
        "DimensionKey": "ReqPerHour",
        "Price": "0.25"
      }
    ]
  }
],
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "StandardEula",
            "Version": "2022-07-14"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {

```

```

        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "CreateOffer",
            "Entity": {
                "Type": "Offer@1.0"
            },
            "ChangeName": "CreateOfferChange",
            "DetailsDocument": {
                "ProductId": "prod-111111111111"
            }
        },
        {
            "ChangeType": "UpdateInformation",
            "Entity": {

```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test private offer for Container product using AWS
Marketplace API Reference Code",
        "Description": "Test private offer for Container product with
contract pricing using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "111111111111"
            ]
        }
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ConfigurableUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "Selector": {
                            "Type": "Duration",
                            "Value": "P12M"
                        },
                        "Constraints": {
                            "MultipleDimensionSelection": "Disallowed",

```

```

        "QuantityConfiguration": "Disallowed"
    },
    "RateCard": [
        {
            "DimensionKey": "ReqPerHour",
            "Price": "0.25"
        }
    ]
}
]
}
]
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},

```

```

    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer (target
buyers)
for my Container product with contract pricing
CAPI-36
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "publish a private offer for my Container product with contract pricing",

```

```
)

return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with contract pricing for an AMI product using an AWS SDK

The following code examples show how to create a private offer with contract pricing for an AMI product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [

```



```

        {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
        }
    ]
}
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ConfigurableUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "Selector": {
                            "Type": "Duration",
                            "Value": "P12M"
                        },
                        "RateCard": [
                            {
                                "DimensionKey": "ReadOnlyUsers",
                                "Price": "220.00"
                            }
                        ]
                    }
                ]
            }
        ]
    }
}

```

```

        "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
        }
    }
]
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "CreateOffer",
            "ChangeName": "CreateOfferChange",
            "Entity": {
                "Type": "Offer@1.0"
            }
        },
    ],
}

```

```

    "DetailsDocument": {
      "ProductId": "prod-111111111111"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {

```

```

        "Type": "CustomEula",
        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
    }
  ]
}
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Contract",
    "Terms": [
      {
        "Type": "ConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P12M"
            },
            "RateCard": [
              {
                "DimensionKey": "ReadOnlyUsers",
                "Price": "220.00"
              }
            ]
          }
        ],
        "Constraints": {

```

```

        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    }
}
]
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
contract pricing for my AMI product
CAPI-35
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

```

```
def main():
    sc.usage_demo(
        change_set, "create private offer with contract pricing for my AMI
product"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with hourly annual pricing and a flexible payment schedule for an AMI product using an AWS SDK

The following code examples show how to create a private offer with hourly annual pricing and a flexible payment schedule for an AMI product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
```

```

    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {

```

```

        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-12-31"
        }
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Usage",
            "Terms": [
                {
                    "Type": "UsageBasedPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.micro",
                                    "Price": "0.17"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    }
]

```



```

        }
      ]
    },
    {
      "Type": "FixedUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "Price": "0.0",
      "Duration": "P365D",
      "Grants": [
        {
          "DimensionKey": "t2.micro",
          "MaxQuantity": 1
        }
      ]
    }
  ]
},
{
  "ChangeType": "UpdateValidityTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "ValidityTerm",
        "AgreementDuration": "P650D"
      }
    ]
  }
},
{
  "ChangeType": "UpdatePaymentScheduleTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",

```

```

        "Schedule": [
            {
                "ChargeDate": "2024-01-01",
                "ChargeAmount": "200.00"
            },
            {
                "ChargeDate": "2024-02-01",
                "ChargeAmount": "170.00"
            }
        ]
    },
    ],
    {
        "ChangeType": "ReleaseOffer",
        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateOfferChange.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "CreateOffer",

```

```

    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [

```

```

        {
            "Type": "LegalTerm",
            "Documents": [
                {
                    "Type": "CustomEula",
                    "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                }
            ]
        }
    ],
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
            {
                "Type": "UsageBasedPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "RateCard": [
                            {
                                "DimensionKey": "t2.micro",
                                "Price": "0.17"
                            }
                        ]
                    }
                ]
            }
        ]
    }
}

```

```

    ]
  },
  {
    "Type": "FixedUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "Price": "0.0",
    "Duration": "P365D",
    "Grants": [
      {
        "DimensionKey": "t2.micro",
        "MaxQuantity": 1
      }
    ]
  }
]
},
{
  "ChangeType": "UpdateValidityTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "ValidityTerm",
        "AgreementDuration": "P650D"
      }
    ]
  }
},
{
  "ChangeType": "UpdatePaymentScheduleTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",
        "Schedule": [

```

```

        {
            "ChargeDate": "2024-01-01",
            "ChargeAmount": "200.00"
        },
        {
            "ChargeDate": "2024-02-01",
            "ChargeAmount": "170.00"
        }
    ]
}
]
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
hourly annual pricing and flexible payment schedule for my AMI product
CAPI-XX
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

```

```
change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Create private offer with hourly annual pricing and flexible payment
        schedule for my AMI product",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with hourly annual pricing for an AMI product using an AWS SDK

The following code examples show how to create a private offer with hourly annual pricing for an AMI product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111",
            "222222222222"
          ]
        }
      }
    },
    {
      "ChangeType": "UpdateLegalTerms",

```



```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2023-12-31"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [

```

```

        {
            "DimensionKey": "t2.micro",
            "Price": "0.17"
        }
    ]
}
],
},
{
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
        {
            "Selector": {
                "Type": "Duration",
                "Value": "P365D"
            },
            "RateCard": [
                {
                    "DimensionKey": "t2.micro",
                    "Price": "220.00"
                }
            ],
            "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
            }
        }
    ]
}
],
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementDuration": "P650D"
            }
        ]
    }
}
]
}

```

```

        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",

```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly annual pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  }
}

```

```

    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Usage",
    "Terms": [
      {
        "Type": "UsageBasedPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "RateCard": [
              {
                "DimensionKey": "t2.micro",
                "Price": "0.17"
              }
            ]
          }
        ]
      },
      {
        "Type": "ConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P365D"
            }
          }
        ]
      }
    ]
  }
}

```

```

        },
        "RateCard": [
            {
                "DimensionKey": "t2.micro",
                "Price": "220.00"
            }
        ],
        "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
        }
    }
]
}
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementDuration": "P650D"
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
hourly annual pricing for my AMI product
CAPI-31
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "create private offer with hourly annual pricing for my AMI
product"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with hourly pricing for an AMI product using an AWS SDK

The following code examples show how to create a private offer with hourly pricing for an AMI product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with hourly pricing for
AmiProduct using AWS Marketplace API Reference Code"
      }
    }
  ]
}
```



```
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
```

```
        "AvailabilityEndDate": "2025-01-01"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
            {
                "Type": "UsageBasedPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "RateCard": [
                            {
                                "DimensionKey": "t2.micro",
                                "Price": "0.15"
                            }
                        ]
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementDuration": "P30D"
            }
        ]
    }
},
},
```

```

    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
    },
  ]
}

```

```

    "DetailsDocument": {
      "Name": "Test private offer for AmiProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with hourly pricing for
AmiProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateAvailability",

```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2025-01-01"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "t2.micro",
                  "Price": "0.15"
                }
              ]
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ValidityTerm",

```

```

        "AgreementDuration": "P30D"
    }
]
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
hourly pricing for my AMI product
CAPI-32
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "create private offer with hourly pricing for my AMI product"
    )

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with subscription pricing for a SaaS product using an AWS SDK

The following code examples show how to create a private offer with subscription pricing for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{  
  "Catalog": "AWSMarketplace",  
  "ChangeSet": [  
    {  
      "ChangeType": "CreateOffer",  
      "Entity": {  
        "Type": "Offer@1.0"  
      },  
      "ChangeName": "CreateOfferChange",  
      "DetailsDocument": {
```

```

        "ProductId": "prod-111111111111"
    }
},
{
    "ChangeType": "UpdateInformation",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "111111111111",
                "222222222222"
            ]
        }
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
            {
                "Type": "UsageBasedPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [

```



```

        {
            "RateCard": [
                {
                    "DimensionKey": "WorkloadSmall",
                    "Price": "0.13"
                },
                {
                    "DimensionKey": "WorkloadMedium",
                    "Price": "0.22"
                }
            ]
        }
    ],
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementDuration": "P30D"
            }
        ]
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {

```

```

        "Type": "CustomEula",
        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
    }
  ]
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111",
            "222222222222"
          ]
        }
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",

```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "WorkloadSmall",
                  "Price": "0.13"
                },
                {
                  "DimensionKey": "WorkloadMedium",
                  "Price": "0.22"
                }
              ]
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ValidityTerm",
          "AgreementDuration": "P30D"
        }
      ]
    }
  }
],
{

```

```

    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2023-12-31"
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
subscription pricing for my SaaS product
CAPI-33
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "Create private offer with subscription pricing for my SaaS
product"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a private offer with tiered contract pricing for a SaaS product using an AWS SDK

The following code examples show how to create a private offer with tiered contract pricing for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",

```

```

        "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdateTargeting",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PositiveTargeting": {
            "BuyerAccounts": [
                "111111111111",
                "222222222222"
            ]
        }
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ConfigurableUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "Selector": {
                            "Type": "Duration",
                            "Value": "P12M"
                        },
                        "RateCard": [
                            {
                                "DimensionKey": "BasicService",
                                "Price": "120.00"
                            },
                            {
                                "DimensionKey": "PremiumService",

```



```

        "Price": "200.00"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Disallowed",
      "QuantityConfiguration": "Disallowed"
    }
  }
]
}
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-12-31"
  }
},

```

```

    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-11111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
    }
  ]
}

```

```
    "DetailsDocument": {
      "Name": "Test private offer for SaaSProduct using AWS Marketplace
API Reference Code",
      "Description": "Test private offer with subscription pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "BasicService",
                  "Price": "120.00"
                }
              ]
            }
          ]
        }
      ]
    }
  }
]
```

```

        },
        {
            "DimensionKey": "PremiumService",
            "Price": "200.00"
        }
    ],
    "Constraints": {
        "MultipleDimensionSelection": "Disallowed",
        "QuantityConfiguration": "Disallowed"
    }
}
]
}
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {

```

```

        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a private offer with
tiered contract pricing for my SaaS product
CAPI-XX
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Create private offer with tiered contract pricing for my SaaS product",
    )

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public free trial offer with subscription pricing for a SaaS product using an AWS SDK

The following code examples show how to create a public free trial offer with subscription pricing for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{  
  "Catalog": "AWSMarketplace",  
  "ChangeSet": [  
    {  
      "ChangeType": "CreateOffer",  
      "Entity": {  
        "Type": "Offer@1.0"  
      },  
      "ChangeName": "CreateOfferChange",  
      "DetailsDocument": {  
        "ProductId": "prod-1111111111111111"  
      }  
    }  
  ]  
}
```

```

    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public free trial offer for SaaSProduct using AWS
Marketplace API Reference Code",
      "Description": "Test public free trial offer with subscription
pricing for SaaSProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Free",
      "Terms": [
        {
          "Type": "FreeTrialPricingTerm",
          "Duration": "P20D",
          "Grants": [
            {
              "DimensionKey": "WorkloadSmall"
            },
            {
              "DimensionKey": "WorkloadMedium"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    }
  }
}

```

```

    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [

```



```

    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateOfferChange",
      "DetailsDocument": {
        "ProductId": "prod-111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test public free trial offer for SaaSProduct using AWS Marketplace API Reference Code",
        "Description": "Test public free trial offer with subscription pricing for SaaSProduct using AWS Marketplace API Reference Code"
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "PricingModel": "Free",
        "Terms": [
          {
            "Type": "FreeTrialPricingTerm",
            "Duration": "P20D",
            "Grants": [
              {
                "DimensionKey": "WorkloadSmall"
              },
              {
                "DimensionKey": "WorkloadMedium"
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```

    ]
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "StandardEula",
            "Version": "2022-07-14"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a public free trial
offer with subscription pricing for SaaS product

```

```
CAPI-13
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Create public free trial offer with subscription pricing for SaaS
product",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a replacement private offer from an existing agreement with contract pricing using an AWS SDK

The following code examples show how to create a replacement private offer from an existing agreement with contract pricing.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateReplacementOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "ChangeName": "CreateReplacementOffer",
      "DetailsDocument": {
        "AgreementId": "agmt-11111111111111111111111111111111"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "Test replacement offer for SaaSProduct using AWS Marketplace API Reference Codes",
        "Description": "Test private replacement offer with contract pricing for SaaSProduct"
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
```

```
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "FixedUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "Price": "0.0",
                "Grants": [
                    {
                        "DimensionKey": "BasicService",
                        "MaxQuantity": 2
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementEndDate": "2024-01-30"
            }
        ]
    }
},
{
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
```

```
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",
        "Schedule": [
            {
                "ChargeDate": "2024-01-01",
                "ChargeAmount": "0"
            }
        ]
    },
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
```

```

        "Entity": {
            "Type": "Offer@1.0",
            "Identifier": "$CreateReplacementOffer.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "CreateReplacementOffer",
            "Entity": {
                "Type": "Offer@1.0"
            },
            "ChangeName": "CreateReplacementOffer",
            "DetailsDocument": {
                "AgreementId": "agmt-11111111111111111111111111111111"
            }
        },
        {
            "ChangeType": "UpdateInformation",
            "Entity": {
                "Type": "Offer@1.0",
                "Identifier": "$CreateReplacementOffer.Entity.Identifier"
            },
            "DetailsDocument": {

```

```
        "Name": "Test replacement offer for SaaSProduct using AWS
Marketplace API Reference Codes",
        "Description": "Test private replacement offer with contract
pricing for SaaSProduct"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "FixedUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "Price": "0.0",
                "Grants": [
                    {
                        "DimensionKey": "BasicService",
                        "MaxQuantity": 2
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateValidityTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "ValidityTerm",
                "AgreementEndDate": "2024-01-30"
            }
        ]
    }
},
```



```
{
  "ChangeType": "UpdatePaymentScheduleTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateReplacementOffer.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",
        "Schedule": [
          {
            "ChargeDate": "2024-01-01",
            "ChargeAmount": "0"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateReplacementOffer.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "StandardEula",
            "Version": "2022-07-14"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateReplacementOffer.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a replacement private
offer
from an existing agreement with contract pricing
CAPI-95
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

```

```
else:
    stringified_change_set = change_set

response = sc.usage_demo(
    stringified_change_set,
    "Create replacement private offer with contract pricing..",
)

return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Describe a public offer using an AWS SDK

The following code examples show how to describe a public offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.catalogapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
```

```
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;

public class DescribeEntity {

    /*
     * Describe my AMI or SaaS or Container product and check if it contains all the
     information I need to know about the product
     */
    public static void main(String[] args) {

        String offerId = args.length > 0 ? args[0] : OFFER_ID;

        DescribeEntityResponse describeEntityResponse =
            getDescribeEntityResponse(offerId);

        ReferenceCodesUtils.formatOutput(describeEntityResponse);
    }

    public static DescribeEntityResponse getDescribeEntityResponse(String offerId) {
        MarketplaceCatalogClient marketplaceCatalogClient =
            MarketplaceCatalogClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        DescribeEntityRequest describeEntityRequest =
            DescribeEntityRequest.builder()
                .catalog(AWS_MP_CATALOG)
                .entityId(offerId)
                .build();

        DescribeEntityResponse describeEntityResponse =
            marketplaceCatalogClient.describeEntity(describeEntityRequest);
        return describeEntityResponse;
    }
}
```

```
}
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) search for offer information in
the AWS Marketplace Catalog
CAPI-29
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

OFFER_ID = "offer-111111111111"

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def get_offer_information(mp_client, entity_id):
    """
```

```
Returns information about a given offer
Args: entity_id str: Entity to return
Returns: dict: Dictionary of offer information
"""

try:
    response = mp_client.describe_entity(
        Catalog="AWSMarketplace",
        EntityId=entity_id,
    )

    return response

except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Offer with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an offer in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")

    pretty_print(get_offer_information(mp_client, OFFER_ID))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Expire a private offer using an AWS SDK

The following code examples show how to set the expiration date of a private offer to a date in the past so that buyers no longer see the offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2023-01-01"
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateOffer",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "Test Private Offer"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create "draft" Private Offer
for any AMI or SAAS product type that can be reviewed internally
before publishing to buyers
CAPI-30
"""

import os

import utils.start_changeset as sc
```



```
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Private offer for AMI product")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List all private offers using an AWS SDK

The following code examples show how to list all private offers.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.List;
```

```
import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.EntitySummary;
import
    software.amazon.awssdk.services.marketplacecatalog.model.EntityTypeFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferAvailabilityEndDateFilter;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferAvailabilityEndDateFilterD
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferBuyerAccountsFilter;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferReleaseDateFilter;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferReleaseDateFilterDateRange
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferTargetingFilter;

public class ListAllPrivateOffers {

    /*
     * List all my private offers and sort or filter them by Offer Publish Date,
     Offer Expiry Date and Buyer IDs
     *
     * OfferTargetingFilter = BuyerAccounts (private offer);
     * OfferBuyerAccountsFilter: Buyer IDs filter
     * OfferAvailabilityEndDateFilter : Offer Expiry Date filter
     * OfferReleaseDateFilter : Offer Publish Date filter
     */
}
```

```
private static MarketplaceCatalogClient marketplaceCatalogClient =
    MarketplaceCatalogClient.builder()
        .httpClient(ApacheHttpClient.builder().build())
        .credentialsProvider(ProfileCredentialsProvider.create())
        .build();

public static void main(String[] args) {

    String offerReleaseDateAfterValue = "2023-01-01T23:59:59Z";
    String offerAvailableEndDateAfterValue = "2040-12-24T23:59:59Z";

    List<EntitySummary> entitySummaryList =
        getEntitySummaryList(offerReleaseDateAfterValue,
            offerAvailableEndDateAfterValue);

    // for each offer id, output the offer detail using DescribeEntity API

    for (EntitySummary entitySummary : entitySummaryList) {
        DescribeEntityRequest describeEntityRequest =
            DescribeEntityRequest.builder()
                .catalog(AWS_MP_CATALOG)
                .entityId(entitySummary.entityId())
                .build();
        DescribeEntityResponse describeEntityResponse =
            marketplaceCatalogClient.describeEntity(describeEntityRequest);
        ReferenceCodesUtils.formatOutput(describeEntityResponse);
    }
}

public static List<EntitySummary> getEntitySummaryList (String
    offerReleaseDateAfterValue, String offerAvailableEndDateAfterValue) {

    EntityTypeFilters entityTypeFilters =
        EntityTypeFilters.builder()
            .offerFilters(OfferFilters.builder()
                .targeting(OfferTargetingFilter.builder()
                    .valueListWithStrings(OFFER_TARGETING_BUYERACCOUNTS)
                    .build())
                .buyerAccounts(OfferBuyerAccountsFilter.builder()
                    .wildCardValue(BUYER_ACCOUNT_ID)
                    .build())
                .availabilityEndDate(OfferAvailabilityEndDateFilter.builder()
                    .dateRange(OfferAvailabilityEndDateFilterDateRange.builder()
```

```
        .afterValue(offerAvailableEndDateAfterValue).build())
        .build())
    .releaseDate(OfferReleaseDateFilter.builder()
        .dateRange(OfferReleaseDateFilterDateRange.builder()
            .afterValue(offerReleaseDateAfterValue)
            .build())
        .build())
    .build())
    .build();

ListEntitiesRequest listEntitiesRequest =
    ListEntitiesRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .entityType(ENTITY_TYPE_OFFER).maxResults(10)
        .entityTypeFilters(entityTypeFilters)
        .nextToken(null)
        .build();

ListEntitiesResponse listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
List<EntitySummary> entitySummaryList = new ArrayList<EntitySummary>();

entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());

while ( listEntitiesResponse.nextToken() != null &&
listEntitiesResponse.nextToken().length() > 0) {
    listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER)
            .maxResults(10)
            .entityTypeFilters(entityTypeFilters)
            .nextToken(listEntitiesResponse.nextToken())
            .build();
    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
    entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());
}

return entitySummaryList;
}
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) for listing offers in the AWS
Marketplace Catalog
CAPI-40
"""
import json
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

# Constants
MAX_RESULTS = 10
CATALOG = "AWSMarketplace"
ENTITY_TYPE = "Offer"

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def list_private_offers(mp_client, return_all_private_offers):
```

```
"""
This method retrieves list of all Private Offers for this account.
"""
entity_summary_list = []
filter_list_param = {
    'OfferFilters': {
        'Targeting': {
            'ValueList': ["BuyerAccounts"]
        }
    }
}
try:
    response = mp_client.list_entities(
        Catalog=CATALOG,
        EntityType=ENTITY_TYPE,
        EntityTypeFilters=filter_list_param,
        MaxResults=MAX_RESULTS
    )
except ClientError as e:
    logger.error("Could not complete list_entities request: %s", e)
    raise

entity_summary_list.extend(response["EntitySummaryList"])
logger.info("Number of results in first iteration: %d " %
len(entity_summary_list))

# Get subsequent pages of results if previous response contained a NextToken
while "NextToken" in response and return_all_private_offers:
    try:
        logger.info("Getting Next Token results: %s " %
response["NextToken"])
        response = mp_client.list_entities(
            Catalog=CATALOG,
            EntityType=ENTITY_TYPE,
            EntityTypeFilters=filter_list_param,
            MaxResults=MAX_RESULTS,
            NextToken=response["NextToken"]
        )
    except ClientError as e:
        logger.error("Could not complete list_entities request: %s", e)
        raise

entity_summary_list.extend(response["EntitySummaryList"])
logger.info(
```

```
        "Number of results in the current iteration: %d "
        % len(response["EntitySummaryList"])
    )

    return entity_summary_list

def get_offer_details(mp_client, offer):
    """
    Describe the details of the Offer.
    """
    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace", EntityId=offer["EntityId"]
        )

        return response
    except ClientError:
        logger.exception("Error: Couldn't get details of the Offer.")
        raise

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Demo - List Private offers.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")

    # Get list of all Offers.
    private_offers = list_private_offers(mp_client, False)
    count = len(private_offers)

    logger.info("Number of Offers: %d " % count)
    offer_counter = 0
    # Display details of each Offer.
    for offer in private_offers:
        print("-" * 88)
        offer_counter += 1
        print("Displaying Offer details for Offer# %d" % offer_counter)
        entity = get_offer_details(mp_client, offer)
        pretty_print(entity)
```

```
print("-" * 88)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List released public and private offers for a specific product ID using an AWS SDK

The following code example shows how to list released public and private offers for a specific product ID.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
```



```
import software.amazon.awssdk.services.marketplacecatalog.model.EntitySummary;
import
    software.amazon.awssdk.services.marketplacecatalog.model.EntityTypeFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferProductIdFilter;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferStateFilter;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferTargetingFilter;

public class ListProductPublicOrPrivateReleasedOffers {

    /*
     * List released Public/Private offers for a specific product id.
     * Example below is to list released public offers.
     * To change to released private offers, change OFFER_TARGETING_NONE (None) to
     * OFFER_TARGETING_BUYERACCOUNTS(BuyerAccounts)
     */
    public static void main(String[] args) {

        List<EntitySummary> entitySummaryList = getEntitySummaryList();
        ReferenceCodesUtils.formatOutput(entitySummaryList);
    }

    public static List<EntitySummary> getEntitySummaryList() {
        MarketplaceCatalogClient marketplaceCatalogClient =
            MarketplaceCatalogClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        // define list entities filters

        EntityTypeFilters entityTypeFilters =
            EntityTypeFilters.builder()
                .offerFilters(OfferFilters.builder()
                    .targeting(OfferTargetingFilter.builder()
                        .valueListWithStrings(OFFER_TARGETING_NONE)
                        .build())
                    .state(OfferStateFilter.builder()
```

```
        .valueListWithStrings(OFFER_STATE_RELEASED)
        .build()
    .productId(OfferProductIdFilter.builder()
        .valueList(PRODUCT_ID)
        .build())
    .build()
    .build();

ListEntitiesRequest listEntitiesRequest =
    ListEntitiesRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .entityType(ENTITY_TYPE_OFFER)
        .maxResults(10)
        .entityTypeFilters(entityTypeFilters)
        .nextToken(null)
        .build();

ListEntitiesResponse listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);

// save all entitySummary of the results into entitySummaryList

List<EntitySummary> entitySummaryList = new ArrayList<EntitySummary>();

entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());

while ( listEntitiesResponse.nextToken() != null &&
listEntitiesResponse.nextToken().length() > 0) {
    listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER)
            .maxResults(10)
            .entityTypeFilters(entityTypeFilters)
            .nextToken(listEntitiesResponse.nextToken())
            .build();
    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
    entitySummaryList.addAll(listEntitiesResponse.entitySummaryList());
}
return entitySummaryList;
}
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update an offer to apply a contract with Pay-As-You-Go pricing using an AWS SDK

The following code examples show how to update an offer to apply a contract with Pay-As-You-Go pricing.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-1111111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
          {
            "Type": "UsageBasedPricingTerm",
```

```
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "RateCard": [
          {
            "DimensionKey": "WorkloadSmall",
            "Price": "0.15"
          },
          {
            "DimensionKey": "WorkloadMedium",
            "Price": "0.25"
          }
        ]
      }
    ],
  },
  {
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P12M"
        },
        "RateCard": [
          {
            "DimensionKey": "BasicService",
            "Price": "150"
          },
          {
            "DimensionKey": "PremiumService",
            "Price": "300"
          }
        ],
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        }
      }
    ]
  }
]
```

```

    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-111111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
          {
            "Type": "UsageBasedPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "RateCard": [
                  {
                    "DimensionKey": "WorkloadSmall",
                    "Price": "0.15"
                  },
                  {
                    "DimensionKey": "WorkloadMedium",
                    "Price": "0.25"
                  }
                ]
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```
    ]
  }
]
},
{
  "Type": "ConfigurableUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "Selector": {
        "Type": "Duration",
        "Value": "P12M"
      },
      "RateCard": [
        {
          "DimensionKey": "BasicService",
          "Price": "150"
        },
        {
          "DimensionKey": "PremiumService",
          "Price": "300"
        }
      ],
      "Constraints": {
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
      }
    }
  ]
}
]
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
```

Shows how to use the AWS SDK for Python (Boto3) to update an offer to apply contract with PAYG pricing

CAPI-21

```
"""
```

```
import os
```

```
import utils.start_changeset as sc
```

```
import utils.stringify_details as sd
```

```
fname = "changeset.json"
```

```
change_set_file = os.path.join(os.path.dirname(__file__), fname)
```

```
change_set = sd.stringify_changeset(change_set_file)
```

```
def main():
```

```
    sc.usage_demo(change_set, "Update offer to apply contract with PAYG pricing")
```

```
if __name__ == "__main__":
```

```
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update an offer to apply hourly annual pricing using an AWS SDK

The following code examples show how to update an offer to apply hourly annual pricing.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
          {
            "Type": "UsageBasedPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "RateCard": [
                  {
                    "DimensionKey": "m5.large",
                    "Price": "0.13"
                  }
                ]
              }
            ]
          },
          {
            "Type": "ConfigurableUpfrontPricingTerm",
            "CurrencyCode": "USD",
```



```

        "RateCards": [
            {
                "Selector": {
                    "Type": "Duration",
                    "Value": "P365D"
                },
                "RateCard": [
                    {
                        "DimensionKey": "m5.large",
                        "Price": "20.03"
                    }
                ],
                "Constraints": {
                    "MultipleDimensionSelection": "Allowed",
                    "QuantityConfiguration": "Allowed"
                }
            }
        ]
    }
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "UpdatePricingTerms",

```

```
"Entity": {
  "Type": "Offer@1.0",
  "Identifier": "offer-111111111111"
},
"DetailsDocument": {
  "PricingModel": "Usage",
  "Terms": [
    {
      "Type": "UsageBasedPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "RateCard": [
            {
              "DimensionKey": "m5.large",
              "Price": "0.13"
            }
          ]
        }
      ]
    },
    {
      "Type": "ConfigurableUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P365D"
          },
          "RateCard": [
            {
              "DimensionKey": "m5.large",
              "Price": "20.03"
            }
          ],
          "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
          }
        }
      ]
    }
  ]
}
```

```
    }  
  }  
]  
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.  
# SPDX-License-Identifier: Apache-2.0  
"""  
Purpose  
Shows how to use the AWS SDK for Python (Boto3) to update an offer to apply  
hourly annual pricing  
CAPI-20  
"""  
  
import os  
  
import utils.start_changeset as sc  
import utils.stringify_details as sd  
  
fname = "changeset.json"  
change_set_file = os.path.join(os.path.dirname(__file__), fname)  
  
change_set = sd.stringify_changeset(change_set_file)  
  
def main():  
    sc.usagem_demo(change_set, "Update offer with hourly annual pricing")  
  
if __name__ == "__main__":  
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update an offer to apply targeting to specific geographic regions using an AWS SDK

The following code examples show how to update an offer to apply targeting to specific geographic regions.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "CountryCodes": [
            "US",
            "ES",
            "FR",
            "AU"
          ]
        }
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateTargeting",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PositiveTargeting": {
          "CountryCodes": [
            "US",
            "ES",
            "FR",
            "AU"
          ]
        }
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
```

Purpose

Shows how to use the AWS SDK for Python (Boto3) to update an offer to apply targeting to certain geographic regions.

CAPI-19

```
"""
```

```
import os
```

```
import utils.start_changeset as sc
import utils.stringify_details as sd
```

```
fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)
```

```
change_set = sd.stringify_changeset(change_set_file)
```

```
def main():
    sc.usage_demo(change_set, "Update offer targeting")
```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update name and description of a public offer using an AWS SDK

The following code examples show how to update name and description of a public offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "LegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-111111111111111"
      },
      "DetailsDocument": {
        "Name": "New offer name",
        "Description": "New offer description"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to update name and description of
my offer
CAPI-18
"""

import os

import utils.start_changeset as sc # type: ignore
```



```
import utils.stringify_details as sd # type: ignore

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Update name and description of my offer")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update the EULA of an offer using an AWS SDK

The following code examples show how to update the EULA of an offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
```

```

"ChangeSet": [
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "offer-111111111111"
    },
    "DetailsDocument": {
      "Name": "New offer name",
      "Description": "New offer description"
    }
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-111111111111"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "LegalTerm",
            "Documents": [
              {

```

```

        "Type": "CustomEula",
        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to update EULA of my offer
CAPI-18
"""

import os

import utils.start_changeset as sc # type: ignore
import utils.stringify_details as sd # type: ignore

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Update EULA of my offer")

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update the expiration date of a private offer to a future date using an AWS SDK

The following code examples show how to update the expiration date of a private offer to a date in the future to give buyers more time to evaluate and accept the offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.


To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-111111111111111"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2026-01-01"
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-1111111111111111"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2026-01-01"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to set expiry date of a private
offer to a date in the future so that my buyers get more time to evaluate and
accept the offer.
CAPI-37
"""

import os

import utils.start_changeset as sc
```

```
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Update offer expiration date")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update the free trial duration of a public free trial offer for a SaaS product using an AWS SDK

The following code examples show how to update the free trial duration of a public free trial offer for a SaaS product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
          {
            "Type": "FreeTrialPricingTerm",
            "Duration": "P21D",
            "Grants": [
              {
                "DimensionKey": "WorkloadSmall"
              },
              {
                "DimensionKey": "WorkloadMedium"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "PricingModel": "Usage",
        "Terms": [
          {
            "Type": "FreeTrialPricingTerm",
            "Duration": "P21D",
            "Grants": [
              {
                "DimensionKey": "WorkloadSmall"
              },
              {
                "DimensionKey": "WorkloadMedium"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to change free trial duration for
my SaaS product by modifying my free trial public offer
CAPI-14
"""

import os
```



```
import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usagem_demo(change_set, "Change free trial duration for SaaS product")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update the refund policy of an offer using an AWS SDK

The following code examples show how to update the refund policy of an offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateSupportTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "SupportTerm",
            "RefundPolicy": "Updated refund policy description"
          }
        ]
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateSupportTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "offer-11111111111111"
      }
    }
  ]
}
```

```

        },
        "DetailsDocument": {
            "Terms": [
                {
                    "Type": "SupportTerm",
                    "RefundPolicy": "Updated refund policy description"
                }
            ]
        }
    ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to update refund policy of my
offer
CAPI-18
"""

import os

import utils.start_changeset as sc # type: ignore
import utils.stringify_details as sd # type: ignore

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Update refund policy of my public offer")

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Products for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Describe an AMI, SaaS, or Container product using an AWS SDK](#)
- [List all AMI, SaaS, or Container products and associated public offers using an AWS SDK](#)

Describe an AMI, SaaS, or Container product using an AWS SDK

The following code examples show how to describe an AMI, SaaS, or Container product and check if it contains all the information you want to know about the product.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.catalogapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
```

```
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;

public class DescribeEntity {

    /**
     * Describe my AMI or SaaS or Container product and check if it contains all the
     information I need to know about the product
     */
    public static void main(String[] args) {

        String offerId = args.length > 0 ? args[0] : OFFER_ID;

        DescribeEntityResponse describeEntityResponse =
            getDescribeEntityResponse(offerId);

        ReferenceCodesUtils.formatOutput(describeEntityResponse);
    }

    public static DescribeEntityResponse getDescribeEntityResponse(String offerId) {
        MarketplaceCatalogClient marketplaceCatalogClient =
            MarketplaceCatalogClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        DescribeEntityRequest describeEntityRequest =
            DescribeEntityRequest.builder()
                .catalog(AWS_MP_CATALOG)
                .entityId(offerId)
                .build();

        DescribeEntityResponse describeEntityResponse =
            marketplaceCatalogClient.describeEntity(describeEntityRequest);
        return describeEntityResponse;
    }
}
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) search for product information in
the AWS Marketplace Catalog
CAPI-28
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

PRODUCT_ID = "prod-111111111111"

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def get_product_information(mp_client, entity_id):
    """
    Returns information about a given product
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of product information
    """

    try:
        response = mp_client.describe_entity(
```

```
        Catalog="AWSMarketplace",
        EntityId=entity_id,
    )

    return response

except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Product with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for a product in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")

    pretty_print(get_product_information(mp_client, PRODUCT_ID))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

List all AMI, SaaS, or Container products and associated public offers using an AWS SDK

The following code examples show how to list all AMI, SaaS, or Container products and associated public offers.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
package com.example.awsmarketplace.catalogapi;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import software.amazon.awssdk.services.marketplacecatalog.model.EntitySummary;
import
    software.amazon.awssdk.services.marketplacecatalog.model.EntityTypeFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.ListEntitiesResponse;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferFilters;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferProductIdFilter;
import software.amazon.awssdk.services.marketplacecatalog.model.OfferStateFilter;
import
    software.amazon.awssdk.services.marketplacecatalog.model.OfferTargetingFilter;

public class ListEntities {

    /*
     * List all my AMI or SaaS or Container products and associated public offers
```



```
*/
public static void main(String[] args) {

    Map<String, List<EntitySummary>> allProductsWithOffers =
    getAllProductsWithOffers();

    ReferenceCodesUtils.formatOutput(allProductsWithOffers);
}

public static Map<String, List<EntitySummary>> getAllProductsWithOffers() {
    MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    Map<String, List<EntitySummary>> allProductsWithOffers = new HashMap<String,
    List<EntitySummary>> ();

    // get all product entities
    List<EntitySummary> productEntityList = new ArrayList<EntitySummary>();

    ListEntitiesRequest listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(PRODUCT_TYPE_AMI)
            .maxResults(10)
            .nextToken(null)
            .build();

    ListEntitiesResponse listEntitiesResponse =
    marketplaceCatalogClient.listEntities(listEntitiesRequest);

    productEntityList.addAll(listEntitiesResponse.entitySummaryList());

    while (listEntitiesResponse.nextToken() != null) {
        listEntitiesRequest =
            ListEntitiesRequest.builder()
                .catalog(AWS_MP_CATALOG)
                .entityType(PRODUCT_TYPE_AMI)
                .maxResults(10)
                .nextToken(listEntitiesResponse.nextToken())
```

```
        .build();
    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
    productEntityList.addAll(listEntitiesResponse.entitySummaryList());
}

// loop through each product entity and get the public released offers
associated using product id filter

for ( EntitySummary productEntitySummary : productEntityList) {
    EntityTypeFilters entityTypeFilters =
        EntityTypeFilters.builder()
            .offerFilters(OfferFilters.builder()
                .targeting(OfferTargetingFilter.builder()
                    .valueListWithStrings(OFFER_TARGETING_NONE)
                    .build())
                .state(OfferStateFilter.builder()
                    .valueListWithStrings(OFFER_STATE_RELEASED)
                    .build())
                .productId(OfferProductIdFilter.builder()
                    .valueList(productEntitySummary.entityId())
                    .build())
                .build()
            ).build();

    listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER)
            .maxResults(10)
            .entityTypeFilters(entityTypeFilters)
            .nextToken(null)
            .build();

    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);

    // save all entitySummary of the results into entitySummaryList

    List<EntitySummary> offerEntitySummaryList = new ArrayList<EntitySummary>();

    offerEntitySummaryList.addAll(listEntitiesResponse.entitySummaryList());
}
```

```
while ( listEntitiesResponse.nextToken() != null &&
listEntitiesResponse.nextToken().length() > 0) {
    listEntitiesRequest =
        ListEntitiesRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityType(ENTITY_TYPE_OFFER)
            .maxResults(10)
            .entityTypeFilters(entityTypeFilters)
            .nextToken(listEntitiesResponse.nextToken())
            .build();
    listEntitiesResponse =
marketplaceCatalogClient.listEntities(listEntitiesRequest);
    offerEntitySummaryList.addAll(listEntitiesResponse.entitySummaryList());
}

// save final results into map; key = product id; value = offer entity summary
list

    allProductsWithOffers.put(productEntitySummary.entityId(),
offerEntitySummaryList);
}
return allProductsWithOffers;
}
}
```

- For API details, see the following topics in *AWS SDK for Java 2.x API Reference*.
 - [DescribeEntity](#)
 - [ListEntities](#)

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to display information about AMI
products and their associated offers in the AWS Marketplace Catalog
CAPI-27
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

MAX_PAGE_RESULTS = 10

try:
    mp_client = boto3.client("marketplace-catalog")
except ClientError as e:
    logger.error("Could not create boto3 client.")
    raise

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def describe_entity(entity_id):
    """
    Returns entity details
    Args: entity_id str: The entity ID of the product or offer
    Returns: dict: The entity details
    """
    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace", EntityId=entity_id
        )
    except ClientError as e:
        logger.error("Could not complete describe_entity request.")
        raise
```

```
# De-stringify the details
response["Details"] = json.loads(response["Details"])

return response

def get_entities(entity_type, visibility=None):
    """
    Returns list of entities for provided entity_type
    Args: entity_type str: Type of entity list to return, in our case AmiProduct
    or Offer
    Returns: list: Abbreviated list of entity information
    """
    EntitySummaryList = []

    # Get the first page of results
    try:
        response = mp_client.list_entities(
            Catalog="AWSMarketplace",
            EntityType=entity_type,
            MaxResults=MAX_PAGE_RESULTS,
        )
    except ClientError as e:
        logger.error("Could not complete list_entities request.")
        raise

    EntitySummaryList.extend(response["EntitySummaryList"])

    # Get subsequent pages of results if previous response contained a NextToken
    while "NextToken" in response:
        try:
            response = mp_client.list_entities(
                Catalog="AWSMarketplace",
                EntityType=entity_type,
                MaxResults=MAX_PAGE_RESULTS,
                NextToken=response["NextToken"],
            )
        except ClientError as e:
            logger.error("Could not complete list_entities request.")
            raise

        EntitySummaryList.extend(response["EntitySummaryList"])
```

```
# if visibility is provided, filter the list to only include entities with
that visibility
if visibility is not None:
    EntitySummaryList = [
        entity for entity in EntitySummaryList if entity["Visibility"] ==
visibility
    ]

    return EntitySummaryList

def get_enhanced_product_list(entity_type):
    """
    Returns an enhanced list of products with product details and offer details
    Args: entity_type str: Type of entity list to return, in our case AmiProduct
    Returns: list: Enhanced list of dictionary objects containing product and
offer details
    """

    product_list = get_entities(entity_type)

    # Loop through product list and append product details to each product
    for product in product_list:
        # appends product details to product dictionary
        product["ProductDetails"] = describe_entity(product["EntityId"])
["Details"]
        # creating an empty list for offer details
        product["OfferDetailsList"] = []

    return product_list

def attach_offer_details(product_list):
    """
    Loops through offer information and appends offer details to product list
    Args: product_list list: List of product dictionaries
    Returns: list: Enhanced list of dictionary objects containing product and
offer details
    """
    offer_list = get_entities("Offer", "Public")

    # Loop through offer list and append offer details to each product
    for offer in offer_list:
        offer["OfferDetails"] = describe_entity(offer["EntityId"])["Details"]
```

```
# Extracts product-id from offer
product_id = offer["OfferDetails"]["ProductId"]

# Determines if product-id referenced in offer matches product-id in
product list
product_dict = next(
    filter(lambda product: product["EntityId"] == product_id,
product_list),
    None,
)

# If product-id matches, appends offer details to product dictionary
if product_dict is not None:
    # logger.info(f"Offer product Id {offer['OfferDetails']['ProductId']}
found in product dictionary. Updating product dictionary with offer details")
    product_dict["OfferDetailsList"].append(offer["OfferDetails"])

else:
    # logger.info("Offer product Id {offer['OfferDetails']['ProductId']}
not found. Skipping offer details update")
    pass

return product_list

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Retrieving products and public offer information...")
    print("-" * 88)

    # Builds a list of products and their details
    product_list = get_enhanced_product_list("AmiProduct")

    # Queries offer information and attaches it to the product list
    product_offer_list = attach_offer_details(product_list)

    pretty_print(product_offer_list)
    return product_offer_list

if __name__ == "__main__":
```

```
usage_demo()
```

- For API details, see the following topics in *AWS SDK for Python (Boto3) API Reference*.
 - [DescribeEntity](#)
 - [ListEntities](#)

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Resale authorization for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Create draft resale authorization for any product type using an AWS SDK](#)
- [Describe a resale authorization using an AWS SDK](#)
- [Publish a one-time resale authorization with a private offer using an AWS SDK](#)
- [Publish multi-use resale authorization with an expiration date using an AWS SDK](#)
- [Publish multi-use resale authorization with an expiration date and a EULA to be sent to the buyer using an AWS SDK](#)
- [Publish multi-use resale authorization with an expiration date and add reseller contract documentation using an AWS SDK](#)
- [Publish multi-use resale authorization with expiration and add a specific buyer account for the resale using an AWS SDK](#)
- [Publish multi-use resale authorization without an expiration date using an AWS SDK](#)
- [Publish multi-use resale authorization without an expiration date and add a custom EULA using an AWS SDK](#)
- [Publish multi-use resale authorization without an expiration date and add reseller contract documentation using an AWS SDK](#)
- [Publish multi-use resale authorization without expiration date and add a specific buyer account for the resale using an AWS SDK](#)
- [Publish one-time resale authorization and add Flexible payment schedule using an AWS SDK](#)

- [Publish one-time resale authorization for any product type and add a EULA using an AWS SDK](#)
- [Publish one-time resale authorization and add a specific buyer account for the resale using an AWS SDK](#)
- [Publish one-time resale authorization for any product type and add reseller contract documentation using an AWS SDK](#)
- [Publish one-time resale authorization for and add whether it is a renewal using an AWS SDK](#)
- [Restrict resale authorization using an AWS SDK](#)
- [Update name and description of one-time or multi-use resale authorization using an AWS SDK](#)

Create draft resale authorization for any product type using an AWS SDK

The following code examples show how to create draft resale authorization for any product type so you can review them internally before publishing to a Channel Partner.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
```

```

        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    }
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    }
  ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a one-time resale authorization on my SaaS/AMI/Container product so my CP
can use that to create Channel Partner Private Offer (CPP0)
CAPI-41
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "draft resale auth")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Describe a resale authorization using an AWS SDK

The following code examples show how to describe a resale authorization.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.catalogapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;

public class DescribeEntity {

    /*
     * Describe my AMI or SaaS or Container product and check if it contains all the
     information I need to know about the product
     */
    public static void main(String[] args) {

        String offerId = args.length > 0 ? args[0] : OFFER_ID;

        DescribeEntityResponse describeEntityResponse =
            getDescribeEntityResponse(offerId);

        ReferenceCodesUtils.formatOutput(describeEntityResponse);
    }
}
```

```
public static DescribeEntityResponse getDescribeEntityResponse(String offerId) {
    MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    DescribeEntityRequest describeEntityRequest =
        DescribeEntityRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityId(offerId)
            .build();

    DescribeEntityResponse describeEntityResponse =
        marketplaceCatalogClient.describeEntity(describeEntityRequest);
    return describeEntityResponse;
}
}
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) search for product information in
the AWS Marketplace Catalog
"""

import json
import logging
```

```
import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

resaleAuthorizationId = "resaleauthz-11111111111111"

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)

def get_product_information(mp_client, entity_id):
    """
    Returns information about a given product
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of product information
    """

    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace",
            EntityId=entity_id,
        )

        return response

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Product with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for a product in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")
```

```
pretty_print(get_product_information(mp_client, resaleAuthorizationId))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [DescribeEntity](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a one-time resale authorization with a private offer using an AWS SDK

The following code examples show how to publish a one-time resale authorization with a private offer so a Channel Partner can use it to create a Channel Partner Private Offer (CPPO).

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      }
    },
  ],
}
```

```

    "DetailsDocument": {
      "ProductId": "prod-111111111111",
      "Name": "TestResaleAuthorization",
      "Description": "Worldwide ResaleAuthorization for Test Product",
      "ResellerAccountId": "111111111111"
    }
  },
  {
    "ChangeType": "ReleaseResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  }
}

```



```

    ]
  }
]
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "OffersMaxQuantity": 1
  }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",

```

```

    "Terms": [
      {
        "Type": "ResaleConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P12M"
            },
            "RateCard": [
              {
                "DimensionKey": "t2.small",
                "Price": "150"
              }
            ],
            "Constraints": {
              "MultipleDimensionSelection": "Allowed",
              "QuantityConfiguration": "Allowed"
            }
          }
        ]
      }
    ],
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerLegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```

        ]
    }
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "OffersMaxQuantity": 1
    }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a one-time resale authorization on my SaaS/AMI/Container product so my CP
can use that to create Channel Partner Private Offer (CPPO)
CAPI-42
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

```

```
response = sc.usage_demo(
    stringified_change_set, "onetime resale auth with private offer"
)

return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization with an expiration date using an AWS SDK

The following code examples show how to publish multi-use resale authorization with an expiration date for an AMI product with hourly annual pricing so a Channel Partner can use it to create a CPPO.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
```

```

    "ChangeName": "ResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0"
    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111",
      "Name": "TestResaleAuthorization",
      "Description": "Worldwide ResaleAuthorization for Test Product",
      "ResellerAccountId": "111111111111"
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",

```

```

        "RateCards": [
            {
                "Selector": {
                    "Type": "Duration",
                    "Value": "P12M"
                },
                "RateCard": [
                    {
                        "DimensionKey": "t2.small",
                        "Price": "150"
                    }
                ],
                "Constraints": {
                    "MultipleDimensionSelection": "Allowed",
                    "QuantityConfiguration": "Allowed"
                }
            }
        ]
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-05-31"
        }
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerLegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/custom-eula.pdf"
              }
            ]
          }
        ]
      }
    }
  ]
}
```



```

    ]
  }
]
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Contract",
    "Terms": [
      {
        "Type": "ResaleConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P12M"
            },
            "RateCard": [
              {
                "DimensionKey": "t2.small",
                "Price": "150"
              }
            ],
            "Constraints": {
              "MultipleDimensionSelection": "Allowed",
              "QuantityConfiguration": "Allowed"
            }
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  }
}

```

```

        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-05-31"
        }
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a multi-use resale authorization with expiry date on my SaaS/AMI product
so my CP can use that to create Channel Partner Private Offer (CPPO)
CAPI-48
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "multi-use resale auth with expiry date")

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization with an expiration date and a EULA to be sent to the buyer using an AWS SDK

The following code examples show how to publish multi-use resale authorization with an expiration date for any product type and add a custom EULA to be sent to the buyer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111",

```

```

        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-05-31"
        }
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Contract",
            "Terms": [
                {
                    "Type": "ResaleConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P12M"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.small",

```

```

        "Price": "150"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  ]
}
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
}
]
}
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2023-05-31"
      }
    }
  ]
}
```

```

    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [

```

```

        {
            "Type": "BuyerLegalTerm",
            "Documents": [
                {
                    "Type": "CustomEula",
                    "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                }
            ]
        }
    ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a multi-use resale authorization with expiry date on my SaaS/AMI/
Container product and add custom EULA to be sent to the buyer
CAPI-56
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "multiuse resale auth with expiry date and custom
EULA")

```



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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization with an expiration date and add reseller contract documentation using an AWS SDK

The following code examples show how to publish multi-use resale authorization with an expiration date for any product type and add reseller contract documentation between the ISV and Channel Partner.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
```

```

        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    }
},
{
    "ChangeType": "ReleaseResaleAuthorization",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {}
},
{
    "ChangeType": "UpdateAvailability",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-05-31"
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "BuyerLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    },
    {
        "Type": "ResaleLegalTerm",

```

```

        "Documents": [
            {
                "Type": "CustomResellerContract",
                "Url": "https://s3.amazonaws.com/aws-mp-standard-
contracts/Standard-Contact-for-AWS-Marketplace-2022-07-14.pdf"}
            ]
        }
    ],
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Contract",
            "Terms": [
                {
                    "Type": "ResaleConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P12M"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.small",
                                    "Price": "150"
                                }
                            ],
                            "Constraints": {
                                "MultipleDimensionSelection": "Allowed",
                                "QuantityConfiguration": "Allowed"
                            }
                        }
                    ]
                }
            ]
        }
    }
}

```

```
]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
```

```

        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "AvailabilityEndDate": "2023-12-31"
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "BuyerLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            },
            {
                "Type": "ResaleLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomResellerContract",
                        "Url": "https://s3.amazonaws.com/aws-mp-standard-
contracts/Standard-Contact-for-AWS-Marketplace-2022-07-14.pdf"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {

```

```

        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ResaleUsageBasedPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "RateCard": [
                            {
                                "DimensionKey": "t2.micro",
                                "Price": "150"
                            }
                        ]
                    }
                ]
            }
        ]
    }
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a multi-use resale authorization with expiry date on my SaaS/AMI/
Container product
and add reseller contract documentation between the ISV and channel partner
CAPI-57
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

```

```
change_set = sd.stringify_changeset(change_set_file)

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "multi use resale auth with contract doc",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization with expiration and add a specific buyer account for the resale using an AWS SDK

The following code examples show how to publish multi-use resale authorization with an expiration date for any product type and add a specific buyer account for the resale.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```



```

    },
    "DetailsDocument": {
      "AvailabilityEndDate": "2023-05-31"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateBuyerTargetingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    }
  }
}

```

```


    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerTargetingTerm",
          "PositiveTargeting": {
            "BuyerAccounts": [
              "111111111111"
            ]
          }
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "AvailabilityEndDate": "2023-05-31"
      }
    }
  ]
}
```

```

    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateBuyerTargetingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [

```

```

        {
            "Type": "BuyerTargetingTerm",
            "PositiveTargeting": {
                "BuyerAccounts": [
                    "111111111111"
                ]
            }
        }
    ],
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "BuyerLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose

```

```
Publish multi-use resale authorization with expiry date for any product type
(AMI/SaaS/Container) and add specific buyer account for the resale
CAPI-82
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Publish multi-use resale authorization with expiry date and add specific
buyer account",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization without an expiration date using an AWS SDK

The following code examples show how to publish multi-use resale authorization without an expiration date for an AMI product with hourly annual pricing so a CP can use that to create a CPPO.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/custom-eula.pdf"
            }
          ]
        }
      ]
    }
  }
}

```



```

    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}

```

```

    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {

```

```

        "Terms": [
            {
                "Type": "BuyerLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",
                        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                    }
                ]
            }
        ]
    }
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a multi-use resale authorization with no expiry date on my SaaS/AMI
product so my CP can use that to create Channel Partner Private Offer (CPPO)
CAPI-52
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usagem_demo(change_set, "multi use resale auth with no expiry date")

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization without an expiration date and add a custom EULA using an AWS SDK

The following code examples show how to publish multi-use resale authorization without an expiration date for any product type and add a custom EULA to be sent to the buyer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{  
  "Catalog": "AWSMarketplace",  
  "ChangeSet": [  
    {  
      "ChangeType": "CreateResaleAuthorization",  
      "ChangeName": "ResaleAuthorization",  
      "Entity": {  
        "Type": "ResaleAuthorization@1.0"  
      },  
      "DetailsDocument": {  
        "ProductId": "prod-11111111111111",  

```

```

        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Contract",
            "Terms": [
                {
                    "Type": "ResaleConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P12M"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.small",
                                    "Price": "150"
                                }
                            ]
                        }
                    ],
                    "Constraints": {
                        "MultipleDimensionSelection": "Allowed",
                        "QuantityConfiguration": "Allowed"
                    }
                }
            ]
        }
    }
}

```

```

    ]
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [

```

```

{
  "ChangeType": "CreateResaleAuthorization",
  "ChangeName": "ResaleAuthorization",
  "Entity": {
    "Type": "ResaleAuthorization@1.0"
  },
  "DetailsDocument": {
    "ProductId": "prod-111111111111",
    "Name": "TestResaleAuthorization",
    "Description": "Worldwide ResaleAuthorization for Test Product",
    "ResellerAccountId": "111111111111"
  }
},
{
  "ChangeType": "ReleaseResaleAuthorization",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {}
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "PricingModel": "Contract",
    "Terms": [
      {
        "Type": "ResaleConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P12M"
            },
            "RateCard": [
              {
                "DimensionKey": "t2.small",
                "Price": "150"
              }
            ]
          }
        ]
      }
    ]
  }
}

```

```

    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
]
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose

```



```
Publish a multi-use resale authorization with no expiry date on my SaaS/AMI/
Container product and add custom EULA to be sent to the buyer
CAPI-58
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "multi use resale auth with no expiry date and custom EULA"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization without an expiration date and add reseller contract documentation using an AWS SDK

The following code examples show how to publish multi-use resale authorization without an expiration date for any product type and add reseller contract documentation between the ISV and Channel Partner.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    },
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  }
}

```

```

        }
      ],
    },
    {
      "Type": "ResaleLegalTerm",
      "Documents": [
        {
          "Type": "CustomResellerContract",
          "Url": "https://s3.amazonaws.com/aws-mp-standard-
contracts/Standard-Contact-for-AWS-Marketplace-2022-07-14.pdf"
        }
      ]
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111",

```

```

        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Contract",
            "Terms": [
                {
                    "Type": "ResaleConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P12M"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.small",
                                    "Price": "150"
                                }
                            ]
                        }
                    ],
                    "Constraints": {
                        "MultipleDimensionSelection": "Allowed",
                        "QuantityConfiguration": "Allowed"
                    }
                }
            ]
        }
    }
}

```

```

    ]
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      },
      {
        "Type": "ResaleLegalTerm",
        "Documents": [
          {
            "Type": "CustomResellerContract",
            "Url": "https://s3.amazonaws.com/aws-mp-standard-
contracts/Standard-Contact-for-AWS-Marketplace-2022-07-14.pdf"
          }
        ]
      }
    ]
  }
}
]
}
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""

```

```
Purpose
Publish a multi-use resale authorization with no expiry date on my SaaS/AMI/
Container product and add reseller contract documentation between the ISV and
channel partner
CAPI-59
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set, "multi use resale auth with no expiry date and contract doc"
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish multi-use resale authorization without expiration date and add a specific buyer account for the resale using an AWS SDK

The following code examples show how to publish multi-use resale authorization without an expiration date for any product type and add a specific buyer account for the resale.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```



```

    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateBuyerTargetingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerTargetingTerm",
          "PositiveTargeting": {
            "BuyerAccounts": [
              "111111111111"
            ]
          }
        }
      ]
    }
  }
}

```

```

    }
  ]
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",

```

```

"ChangeSet": [
  {
    "ChangeType": "CreateResaleAuthorization",
    "ChangeName": "ResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0"
    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111",
      "Name": "TestResaleAuthorization",
      "Description": "Worldwide ResaleAuthorization for Test Product",
      "ResellerAccountId": "111111111111"
    }
  },
  {
    "ChangeType": "ReleaseResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ]
            }
          ]
        }
      ]
    }
  }
]

```

```

        }
    ],
    "Constraints": {
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    }
}
]
}
],
},
{
    "ChangeType": "UpdateBuyerTargetingTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "BuyerTargetingTerm",
                "PositiveTargeting": {
                    "BuyerAccounts": [
                        "111111111111"
                    ]
                }
            }
        ]
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "BuyerLegalTerm",
                "Documents": [
                    {
                        "Type": "CustomEula",

```

```

        "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
    }
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish multi-use resale authorization without expiry date for any product type
(AMI/SaaS/Container) and add specific buyer account for the resale
CAPI-83
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "publish multi-use resale authorization without expiry date and add
specific buyer account",
    )

```

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish one-time resale authorization and add Flexible payment schedule using an AWS SDK

The following code examples show how to publish one-time resale authorization for any product type and add Flexible payment schedule.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
```

```

        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    }
},
{
    "ChangeType": "ReleaseResaleAuthorization",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {}
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ResaleFixedUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "Price": "0.00",
                "Duration": "P12M",
                "Grants": [
                    {
                        "DimensionKey": "Users",
                        "MaxQuantity": 10
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [

```

```

        {
            "Type": "ResalePaymentScheduleTerm",
            "CurrencyCode": "USD",
            "Schedule": [
                {
                    "ChargeDate": "2023-09-01",
                    "ChargeAmount": "200.00"
                },
                {
                    "ChargeDate": "2023-12-01",
                    "ChargeAmount": "250.00"
                }
            ]
        }
    ],
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "AvailabilityEndDate": "2023-06-30",
            "OffersMaxQuantity": 1
        }
    },
    {
        "ChangeType": "UpdateLegalTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "Terms": [
                {
                    "Type": "BuyerLegalTerm",
                    "Documents": [
                        {
                            "Type": "CustomEula",
                            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
                        }
                    ]
                }
            ]
        }
    }
}

```



```

    ]
  }
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}

```

```

    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleFixedUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "Price": "0.00",
          "Duration": "P12M",
          "Grants": [
            {
              "DimensionKey": "Users",
              "MaxQuantity": 10
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdatePaymentScheduleTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "ResalePaymentScheduleTerm",
          "CurrencyCode": "USD",
          "Schedule": [
            {
              "ChargeDate": "2023-09-01",
              "ChargeAmount": "200.00"
            },
            {
              "ChargeDate": "2023-12-01",

```

```

        "ChargeAmount": "250.00"
      }
    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "AvailabilityEndDate": "2023-06-30",
    "OffersMaxQuantity": 1
  }
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      }
    ]
  }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish one-time resale authorization for any product type (AMI/SaaS/Container)
and add Flexible payment schedule
CAPI-78
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "onetime resale auth with flexible payment
    schedule")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish one-time resale authorization for any product type and add a EULA using an AWS SDK

The following code examples show how to publish one-time resale authorization for any product type and add a custom EULA to be sent to the buyer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}
```

```

    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "OffersMaxQuantity": 1
      }
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
          {
            "Type": "ResaleConfigurableUpfrontPricingTerm",
            "CurrencyCode": "USD",
            "RateCards": [
              {
                "Selector": {
                  "Type": "Duration",
                  "Value": "P12M"
                },
                "RateCard": [
                  {
                    "DimensionKey": "t2.small",
                    "Price": "150"
                  }
                ],
                "Constraints": {
                  "MultipleDimensionSelection": "Allowed",
                  "QuantityConfiguration": "Allowed"
                }
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```

    },
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerLegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
              }
            ]
          }
        ]
      }
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",

```

```

    "ChangeName": "ResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0"
    },
    "DetailsDocument": {
      "ProductId": "prod-111111111111",
      "Name": "TestResaleAuthorization",
      "Description": "Worldwide ResaleAuthorization for Test Product",
      "ResellerAccountId": "111111111111"
    }
  },
  {
    "ChangeType": "ReleaseResaleAuthorization",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdateAvailability",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "OffersMaxQuantity": 1
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {

```


Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a one-time resale authorization on my SaaS/AMI/Container product and add
custom EULA to be sent to the buyer
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "onetime resale auth with custom EULA")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish one-time resale authorization and add a specific buyer account for the resale using an AWS SDK

The following code examples show how to publish one-time resale authorization for any product type and add a specific buyer account for the resale.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/custom-eula.pdf"
            }
          ]
        }
      ]
    }
  }
}

```

```

    }
  ]
}
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "OffersMaxQuantity": "1"
  }
},
{
  "ChangeType": "UpdateBuyerTargetingTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerTargetingTerm",
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111"
          ]
        }
      }
    ]
  }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdatePricingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "PricingModel": "Contract",

```

```

    "Terms": [
      {
        "Type": "ResaleConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P12M"
            },
            "RateCard": [
              {
                "DimensionKey": "t2.small",
                "Price": "150"
              }
            ],
            "Constraints": {
              "MultipleDimensionSelection": "Allowed",
              "QuantityConfiguration": "Allowed"
            }
          }
        ]
      }
    ],
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerLegalTerm",
            "Documents": [
              {
                "Type": "CustomEula",
                "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
              }
            ]
          }
        ]
      }
    }
  ]
}

```

```

    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "OffersMaxQuantity": 1
  }
},
{
  "ChangeType": "UpdateBuyerTargetingTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerTargetingTerm",
        "PositiveTargeting": {
          "BuyerAccounts": [
            "111111111111"
          ]
        }
      }
    ]
  }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose

```



```
Publish one-time resale authorization for any product type (AMI/SaaS/Container)
and add specific buyer account for the resale
CAPI-81
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "onetime resale authorization for specific buyer
account")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.


For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish one-time resale authorization for any product type and add reseller contract documentation using an AWS SDK

The following code examples show how to publish one-time resale authorization for any product type and add reseller contract documentation between the ISV and Channel Partner.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "ReleaseResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateAvailability",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "OffersMaxQuantity": 1
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ResaleConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P12M"
              },
              "RateCard": [
                {
                  "DimensionKey": "t2.small",
                  "Price": "150"
                }
              ],
              "Constraints": {
                "MultipleDimensionSelection": "Allowed",
                "QuantityConfiguration": "Allowed"
              }
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "ResaleAuthorization@1.0",
      "Identifier": "$ResaleAuthorization.Entity.Identifier"
    }
  }
}

```

```

    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "BuyerLegalTerm",
          "Documents": [
            {
              "Type": "CustomEula",
              "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
            }
          ]
        }
      ]
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-11111111111111",

```

```

        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
    },
    {
        "ChangeType": "ReleaseResaleAuthorization",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {}
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "OffersMaxQuantity": 1
        }
    },
    {
        "ChangeType": "UpdatePricingTerms",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "PricingModel": "Contract",
            "Terms": [
                {
                    "Type": "ResaleConfigurableUpfrontPricingTerm",
                    "CurrencyCode": "USD",
                    "RateCards": [
                        {
                            "Selector": {
                                "Type": "Duration",
                                "Value": "P12M"
                            },
                            "RateCard": [
                                {
                                    "DimensionKey": "t2.small",

```

```

        "Price": "150"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
]
}
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "BuyerLegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/sample-bucket/
custom-eula.pdf"
          }
        ]
      },
      {
        "Type": "ResaleLegalTerm",
        "Documents": [
          {
            "Type": "CustomResellerContract",
            "Url": "https://s3.amazonaws.com/aws-mp-standard-
contracts/Standard-Contract-for-AWS-Marketplace-2022-07-14.pdf"
          }
        ]
      }
    ]
  }
}
}

```

```
]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish a one-time resale authorization on my SaaS/AMI/Container product and add
reseller contract documentation between the ISV and channel partner
CAPI-47
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "onetime resale auth with reseller contract doc")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish one-time resale authorization for and add whether it is a renewal using an AWS SDK

The following code examples show how to publish one-time resale authorization for any product type and add whether it is a renewal.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "UpdateBuyerTargetingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
```



```

        "Terms": [
            {
                "Type": "BuyerTargetingTerm",
                "PositiveTargeting": {
                    "BuyerAccounts": [
                        "222222222222"
                    ]
                }
            }
        ],
    },
    {
        "ChangeType": "UpdateAvailability",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "OffersMaxQuantity": 1
        }
    },
    {
        "ChangeType": "UpdateInformation",
        "Entity": {
            "Type": "ResaleAuthorization@1.0",
            "Identifier": "$ResaleAuthorization.Entity.Identifier"
        },
        "DetailsDocument": {
            "Name": "TestResaleAuthorization",
            "Description": "Worldwide ResaleAuthorization for Test Product",
            "PreExistingBuyerAgreement": {
                "AcquisitionChannel": "AwsMarketplace",
                "PricingModel": "Contract"
            }
        }
    }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateResaleAuthorization",
      "ChangeName": "ResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0"
      },
      "DetailsDocument": {
        "ProductId": "prod-111111111111",
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product",
        "ResellerAccountId": "111111111111"
      }
    },
    {
      "ChangeType": "UpdateBuyerTargetingTerms",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "$ResaleAuthorization.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "BuyerTargetingTerm",
            "PositiveTargeting": {
              "BuyerAccounts": [
                "222222222222"
              ]
            }
          }
        ]
      }
    }
  ]
}
```

```

    ]
  }
},
{
  "ChangeType": "UpdateAvailability",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "OffersMaxQuantity": 1
  }
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "ResaleAuthorization@1.0",
    "Identifier": "$ResaleAuthorization.Entity.Identifier"
  },
  "DetailsDocument": {
    "Name": "TestResaleAuthorization",
    "Description": "Worldwide ResaleAuthorization for Test Product",
    "PreExistingBuyerAgreement": {
      "AcquisitionChannel": "AwsMarketplace",
      "PricingModel": "Contract"
    }
  }
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish one-time resale authorization for any product type (AMI/SaaS/Container)
and add whether it is renewal or not
CAPI-90
"""

```

```
import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(stringified_change_set, "onetime resale auth
renewal")

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Restrict resale authorization using an AWS SDK

The following code examples show how to restrict resale authorization.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictResaleAuthorization",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-111111111111111"
      },
      "DetailsDocument": {}
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
```

```

    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "RestrictResaleAuthorization",
            "Entity": {
                "Type": "ResaleAuthorization@1.0",
                "Identifier": "resaleauthz-11111111111111"
            },
            "DetailsDocument": {}
        }
    ]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Restrict a authorization for any product type (AMI/SaaS/Container)
CAPI-84
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "Restrict resale authorization")

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update name and description of one-time or multi-use resale authorization using an AWS SDK

The following code examples show how to update name and description of one-time or multi-use resale authorization before publishing for any product type.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to RunChangesets in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-11111111111111"
      },
      "DetailsDocument": {
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product"
      }
    }
  ]
}
```

```
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "ResaleAuthorization@1.0",
        "Identifier": "resaleauthz-11111111111111"
      },
      "DetailsDocument": {
        "Name": "TestResaleAuthorization",
        "Description": "Worldwide ResaleAuthorization for Test Product"
      }
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Update name/description of one-time or multi-use resale authorization before
publishing for any product type (AMI/SaaS/Container)
```



```
CAPI-77
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "update name and description of one-time or multi-use resale
        authorization before publishing",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

SaaS products for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Create a draft SaaS product with a draft public offer using an AWS SDK](#)
- [Create a public or limited SaaS product and public offer with contract pricing using an AWS SDK](#)
- [Create a public or limited SaaS product and public offer with contract with Pay-As-You-Go pricing using an AWS SDK](#)

- [Create a public or limited SaaS product and public offer with subscription pricing using an AWS SDK](#)
- [Publish a SaaS product and associated public offer using an AWS SDK](#)
- [Publish a SaaS product and associated public offer from an existing draft using an AWS SDK](#)
- [Update dimensions on an AMI or SaaS product using an AWS SDK](#)

Create a draft SaaS product with a draft public offer using an AWS SDK

The following code examples show how to create a draft SaaS product with a draft public offer.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product"
      }
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
```

```

        "Type": "Offer@1.0"
    },
    "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier",
        "Name": "Test Offer"
    }
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier",
        "Name": "Test Offer"
      }
    }
  ]
}

```

```

    }
  }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create an SaaS draft product
with a draft public offer.
CAPI-04
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "Create a draft saas product with a draft public offer",
    )

    return response

if __name__ == "__main__":
    main()

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited SaaS product and public offer with contract pricing using an AWS SDK

The following code examples show how to create a public or limited SaaS product and public offer with contract pricing. This example creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "ChangeName": "CreateProductChange",
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      }
    }
  ]
}
```

```

    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Data Catalogs"
      ],
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awssmp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  }
},
{
  "ChangeType": "AddDeliveryOptions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "DeliveryOptions": [

```

```

        {
            "Details": {
                "SaaSUrlDeliveryOptionDetails": {
                    "FulfillmentUrl": "https://sample.amazonaws.com/
sample-saas-fulfillment-url"
                }
            }
        }
    ],
},
{
    "ChangeType": "AddDimensions",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
        {
            "Key": "BasicService",
            "Description": "Basic Service",
            "Name": "Basic Service",
            "Types": [
                "Entitled"
            ],
            "Unit": "Units"
        },
        {
            "Key": "PremiumService",
            "Description": "Premium Service",
            "Name": "Premium Service",
            "Types": [
                "Entitled"
            ],
            "Unit": "Units"
        }
    ]
},
{
    "ChangeType": "ReleaseProduct",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
},

```

```

    "DetailsDocument": {}
  },
  {
    "ChangeType": "CreateOffer",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "ChangeName": "CreateOfferChange",
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
      "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P1M"
              },
              "RateCard": [

```



```

        {
            "DimensionKey": "BasicService",
            "Price": "20"
        },
        {
            "DimensionKey": "PremiumService",
            "Price": "25"
        }
    ],
    "Constraints": {
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    }
},
{
    "Selector": {
        "Type": "Duration",
        "Value": "P12M"
    },
    "RateCard": [
        {
            "DimensionKey": "BasicService",
            "Price": "150"
        },
        {
            "DimensionKey": "PremiumService",
            "Price": "300"
        }
    ],
    "Constraints": {
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
    }
}
]
}
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    }
}

```

```


    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "SupportTerm",
          "RefundPolicy": "Absolutely no refund, period."
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "ChangeName": "CreateProductChange",
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
          "Sample highlight"
        ],
        "SearchKeywords": [
          "Sample keyword"
        ],
        "Categories": [
          "Data Catalogs"
        ],
        "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
        "VideoUrls": [
```

```

        "https://sample.amazonaws.com/awssmp-video-1"
    ],
    "AdditionalResources": []
  }
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111",
        "222222222222"
      ]
    }
  }
},
{
  "ChangeType": "AddDeliveryOptions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "DeliveryOptions": [
      {
        "Details": {
          "SaaSUrlDeliveryOptionDetails": {
            "FulfillmentUrl": "https://sample.amazonaws.com/
sample-saas-fulfillment-url"
          }
        }
      ]
    }
  }
},
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  }
}

```

```

    },
    "DetailsDocument": [
      {
        "Key": "BasicService",
        "Description": "Basic Service",
        "Name": "Basic Service",
        "Types": [
          "Entitled"
        ],
        "Unit": "Units"
      },
      {
        "Key": "PremiumService",
        "Description": "Premium Service",
        "Name": "Premium Service",
        "Types": [
          "Entitled"
        ],
        "Unit": "Units"
      }
    ]
  },
  {
    "ChangeType": "ReleaseProduct",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "CreateOffer",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "ChangeName": "CreateOfferChange",
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",

```

```

        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",
        "Terms": [
            {
                "Type": "ConfigurableUpfrontPricingTerm",
                "CurrencyCode": "USD",
                "RateCards": [
                    {
                        "Selector": {
                            "Type": "Duration",
                            "Value": "P1M"
                        },
                        "RateCard": [
                            {
                                "DimensionKey": "BasicService",
                                "Price": "20"
                            },
                            {
                                "DimensionKey": "PremiumService",
                                "Price": "25"
                            }
                        ]
                    }
                ],
                "Constraints": {
                    "MultipleDimensionSelection": "Allowed",
                    "QuantityConfiguration": "Allowed"
                }
            },
            {
                "Selector": {

```



```

    "ChangeType": "UpdateSupportTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "SupportTerm",
          "RefundPolicy": "Absolutely no refund, period."
        }
      ]
    }
  },
  {
    "ChangeType": "UpdateRenewalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "RenewalTerm"
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseOffer",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  }
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
```



```
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a
public or limited SaaS product and public offer with contract pricing and
standard EULA
CAPI-11
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(
        change_set,
        "Create a limited saas product with a public offer with contract
pricing",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited SaaS product and public offer with contract with Pay-As-You-Go pricing using an AWS SDK

The following code examples show how to create a public or limited SaaS product and public offer with a contract with Pay-As-You-Go pricing. This example creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "ChangeName": "CreateProductChange",
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
```



```

    }
  }
]
},
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": [
    {
      "Key": "BasicService",
      "Description": "Basic Service",
      "Name": "Basic Service",
      "Types": [
        "Entitled"
      ],
      "Unit": "Units"
    },
    {
      "Key": "PremiumService",
      "Description": "Premium Service",
      "Name": "Premium Service",
      "Types": [
        "Entitled"
      ],
      "Unit": "Units"
    },
    {
      "Key": "WorkloadSmall",
      "Description": "Workload: Per medium instance",
      "Name": "Workload: Per medium instance",
      "Types": [
        "ExternallyMetered"
      ],
      "Unit": "Units"
    },
    {
      "Key": "WorkloadMedium",
      "Description": "Workload: Per large instance",
      "Name": "Workload: Per large instance",
      "Types": [

```

```

        "ExternallyMetered"
    ],
    "Unit": "Units"
  }
]
},
{
  "ChangeType": "ReleaseProduct",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {}
},
{
  "ChangeType": "CreateOffer",
  "Entity": {
    "Type": "Offer@1.0"
  },
  "ChangeName": "CreateOfferChange",
  "DetailsDocument": {
    "ProductId": "$CreateProductChange.Entity.Identifier"
  }
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
    "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {

```

```
"PricingModel": "Contract",
"Terms": [
  {
    "Type": "UsageBasedPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "RateCard": [
          {
            "DimensionKey": "WorkloadSmall",
            "Price": "0.15"
          },
          {
            "DimensionKey": "WorkloadMedium",
            "Price": "0.25"
          }
        ]
      }
    ]
  },
  {
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P12M"
        },
        "RateCard": [
          {
            "DimensionKey": "BasicService",
            "Price": "150"
          },
          {
            "DimensionKey": "PremiumService",
            "Price": "300"
          }
        ],
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        }
      }
    ]
  }
]
```

```

    ]
  }
]
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "StandardEula",
            "Version": "2022-07-14"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateSupportTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "Absolutely no refund, period."
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",

```

```

        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "ChangeName": "CreateProductChange",
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",
        "Highlights": [
          "Sample highlight"
        ]
      }
    }
  ]
}

```



```
    ],
    "SearchKeywords": [
      "Sample keyword"
    ],
    "Categories": [
      "Data Catalogs"
    ],
    "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
    "VideoUrls": [
      "https://sample.amazonaws.com/awsmvp-video-1"
    ],
    "AdditionalResources": []
  }
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111",
        "222222222222"
      ]
    }
  }
},
{
  "ChangeType": "AddDeliveryOptions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "DeliveryOptions": [
      {
        "Details": {
          "SaaSUrlDeliveryOptionDetails": {
            "FulfillmentUrl": "https://sample.amazonaws.com/sample-saas-fulfillment-url"
          }
        }
      ]
    }
  }
}
```

```

    }
  ]
}
},
{
  "ChangeType": "AddDimensions",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": [
    {
      "Key": "BasicService",
      "Description": "Basic Service",
      "Name": "Basic Service",
      "Types": [
        "Entitled"
      ],
      "Unit": "Units"
    },
    {
      "Key": "PremiumService",
      "Description": "Premium Service",
      "Name": "Premium Service",
      "Types": [
        "Entitled"
      ],
      "Unit": "Units"
    },
    {
      "Key": "WorkloadSmall",
      "Description": "Workload: Per medium instance",
      "Name": "Workload: Per medium instance",
      "Types": [
        "ExternallyMetered"
      ],
      "Unit": "Units"
    },
    {
      "Key": "WorkloadMedium",
      "Description": "Workload: Per large instance",
      "Name": "Workload: Per large instance",
      "Types": [
        "ExternallyMetered"
      ]
    }
  ]
}

```

```

        ],
        "Unit": "Units"
    }
]
},
{
    "ChangeType": "ReleaseProduct",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
},
{
    "ChangeType": "CreateOffer",
    "Entity": {
        "Type": "Offer@1.0"
    },
    "ChangeName": "CreateOfferChange",
    "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier"
    }
},
{
    "ChangeType": "UpdateInformation",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "PricingModel": "Contract",

```

```
"Terms": [
  {
    "Type": "UsageBasedPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "RateCard": [
          {
            "DimensionKey": "WorkloadSmall",
            "Price": "0.15"
          },
          {
            "DimensionKey": "WorkloadMedium",
            "Price": "0.25"
          }
        ]
      }
    ]
  },
  {
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P12M"
        },
        "RateCard": [
          {
            "DimensionKey": "BasicService",
            "Price": "150"
          },
          {
            "DimensionKey": "PremiumService",
            "Price": "300"
          }
        ],
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        }
      }
    ]
  }
]
```

```

    }
  ]
}
},
{
  "ChangeType": "UpdateLegalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "StandardEula",
            "Version": "2022-07-14"
          }
        ]
      }
    ]
  }
},
{
  "ChangeType": "UpdateSupportTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "Absolutely no refund, period."
      }
    ]
  }
},
{
  "ChangeType": "UpdateRenewalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  }
}

```

```

    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "RenewalTerm"
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a
public or limited SaaS product and public offer with contract with PAYG pricing
and standard EULA
CAPI-10
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

```

```
def main():
    sc.usage_demo(
        change_set,
        "Create limited SaaS product with public offer with contract with payg
pricing",
    )

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Create a public or limited SaaS product and public offer with subscription pricing using an AWS SDK

The following code examples show how to create a public or limited SaaS product and public offer with subscription pricing. This examples creates either a standard or custom EULA.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
```

```
{
  "ChangeType": "CreateProduct",
  "Entity": {
    "Type": "SaaSProduct@1.0"
  },
  "ChangeName": "CreateProductChange",
  "DetailsDocument": {}
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "ProductTitle": "Sample product",
    "ShortDescription": "Brief description",
    "LongDescription": "Detailed description",
    "Highlights": [
      "Sample highlight"
    ],
    "SearchKeywords": [
      "Sample keyword"
    ],
    "Categories": [
      "Data Catalogs"
    ],
    "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
    "VideoUrls": [
      "https://sample.amazonaws.com/awsmvp-video-1"
    ],
    "AdditionalResources": []
  }
},
{
  "ChangeType": "UpdateTargeting",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "PositiveTargeting": {
      "BuyerAccounts": [
        "111111111111",

```



```

                "222222222222"
            ]
        }
    },
    {
        "ChangeType": "AddDeliveryOptions",
        "Entity": {
            "Type": "SaaSProduct@1.0",
            "Identifier": "$CreateProductChange.Entity.Identifier"
        },
        "DetailsDocument": {
            "DeliveryOptions": [
                {
                    "Details": {
                        "SaaSUrlDeliveryOptionDetails": {
                            "FulfillmentUrl": "https://sample.amazonaws.com/
sample-saas-fulfillment-url"
                        }
                    }
                }
            ]
        }
    },
    {
        "ChangeType": "AddDimensions",
        "Entity": {
            "Type": "SaaSProduct@1.0",
            "Identifier": "$CreateProductChange.Entity.Identifier"
        },
        "DetailsDocument": [
            {
                "Key": "WorkloadSmall",
                "Description": "Workload: Per medium instance",
                "Name": "Workload: Per medium instance",
                "Types": [
                    "ExternallyMetered"
                ],
                "Unit": "Units"
            },
            {
                "Key": "WorkloadMedium",
                "Description": "Workload: Per large instance",
                "Name": "Workload: Per large instance",
            }
        ]
    }
]

```

```

        "Types": [
            "ExternallyMetered"
        ],
        "Unit": "Units"
    }
]
},
{
    "ChangeType": "ReleaseProduct",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
},
{
    "ChangeType": "CreateOffer",
    "Entity": {
        "Type": "Offer@1.0"
    },
    "ChangeName": "CreateOfferChange",
    "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier"
    }
},
{
    "ChangeType": "UpdateInformation",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
        "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },

```

```

    "DetailsDocument": {
      "PricingModel": "Usage",
      "Terms": [
        {
          "Type": "UsageBasedPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "RateCard": [
                {
                  "DimensionKey": "WorkloadSmall",
                  "Price": "0.15"
                },
                {
                  "DimensionKey": "WorkloadMedium",
                  "Price": "0.25"
                }
              ]
            }
          ]
        }
      ]
    },
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "LegalTerm",
            "Documents": [
              {
                "Type": "StandardEula",
                "Version": "2022-07-14"
              }
            ]
          }
        ]
      }
    },
  ],
}

```

```

    {
      "ChangeType": "UpdateSupportTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "SupportTerm",
            "RefundPolicy": "Absolutely no refund, period."
          }
        ]
      }
    },
    {
      "ChangeType": "ReleaseOffer",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {}
    }
  ]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {

```

```

    "ChangeType": "CreateProduct",
    "Entity": {
      "Type": "SaaSProduct@1.0"
    },
    "ChangeName": "CreateProductChange",
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Data Catalogs"
      ],
      "LogoUrl": "https://s3.amazonaws.com/logos/sample.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awssmp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "UpdateTargeting",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PositiveTargeting": {
        "BuyerAccounts": [
          "111111111111",
          "222222222222"
        ]
      }
    }
  }
}

```

```

        ]
      }
    },
  },
  {
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "DeliveryOptions": [
        {
          "Details": {
            "SaaSUrlDeliveryOptionDetails": {
              "FulfillmentUrl": "https://sample.amazonaws.com/
sample-saas-fulfillment-url"
            }
          }
        }
      ]
    }
  },
  {
    "ChangeType": "AddDimensions",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
      {
        "Key": "WorkloadSmall",
        "Description": "Workload: Per medium instance",
        "Name": "Workload: Per medium instance",
        "Types": [
          "ExternallyMetered"
        ],
        "Unit": "Units"
      },
      {
        "Key": "WorkloadMedium",
        "Description": "Workload: Per large instance",
        "Name": "Workload: Per large instance",
        "Types": [

```

```

        "ExternallyMetered"
    ],
    "Unit": "Units"
  }
]
},
{
  "ChangeType": "ReleaseProduct",
  "Entity": {
    "Type": "SaaSProduct@1.0",
    "Identifier": "$CreateProductChange.Entity.Identifier"
  },
  "DetailsDocument": {}
},
{
  "ChangeType": "CreateOffer",
  "Entity": {
    "Type": "Offer@1.0"
  },
  "ChangeName": "CreateOfferChange",
  "DetailsDocument": {
    "ProductId": "$CreateProductChange.Entity.Identifier"
  }
},
{
  "ChangeType": "UpdateInformation",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Name": "Test public offer for SaaSProduct using AWS Marketplace
API Reference Code",
    "Description": "Test public offer with contract pricing for
SaaSProduct using AWS Marketplace API Reference Code"
  }
},
{
  "ChangeType": "UpdatePricingTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {

```

```

    "PricingModel": "Usage",
    "Terms": [
      {
        "Type": "UsageBasedPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "RateCard": [
              {
                "DimensionKey": "WorkloadSmall",
                "Price": "0.15"
              },
              {
                "DimensionKey": "WorkloadMedium",
                "Price": "0.25"
              }
            ]
          }
        ]
      }
    ],
  },
  {
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "LegalTerm",
          "Documents": [
            {
              "Type": "StandardEula",
              "Version": "2022-07-14"
            }
          ]
        }
      ]
    }
  },
  {

```



```

    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",
                "RefundPolicy": "Absolutely no refund, period."
            }
        ]
    }
},
{
    "ChangeType": "ReleaseOffer",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to create a
public or limited SaaS product and public offer with subscription(usage) pricing
and standard EULA
CAPI-09
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

```

```
fname = "changeset.json"
change_set_file = os.path.join(os.path.dirname(__file__), fname)

change_set = sd.stringify_changeset(change_set_file)

def main():
    sc.usage_demo(change_set, "public saas public offer with subscription
pricing")

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a SaaS product and associated public offer using an AWS SDK

The following code examples show how to publish a SaaS product and associated public offer. The product will be in a limited state by default.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
```

```
"ChangeSet": [
  {
    "ChangeType": "CreateProduct",
    "ChangeName": "CreateProductChange",
    "Entity": {
      "Type": "SaaSProduct@1.0"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "ProductTitle": "Sample product",
      "ShortDescription": "Brief description",
      "LongDescription": "Detailed description",
      "Highlights": [
        "Sample highlight"
      ],
      "SearchKeywords": [
        "Sample keyword"
      ],
      "Categories": [
        "Data Catalogs"
      ],
      "LogoUrl": "https://bucketname.s3.amazonaws.com/logo.png",
      "VideoUrls": [
        "https://sample.amazonaws.com/awsmp-video-1"
      ],
      "AdditionalResources": []
    }
  },
  {
    "ChangeType": "AddDimensions",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
      {
        "Key": "BasicService",
```

```

        "Description": "Basic Service",
        "Name": "Basic Service",
        "Types": [
            "Entitled"
        ],
        "Unit": "Units"
    },
    {
        "Key": "PremiumService",
        "Description": "Premium Service",
        "Name": "Premium Service",
        "Types": [
            "Entitled"
        ],
        "Unit": "Units"
    }
]
},
{
    "ChangeType": "AddDeliveryOptions",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "DeliveryOptions": [
            {
                "Details": {
                    "SaaSUrlDeliveryOptionDetails": {
                        "FulfillmentUrl": "https://www.aws.amazon.com/
marketplace/management"
                    }
                }
            }
        ]
    }
},
{
    "ChangeType": "ReleaseProduct",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}

```

```
    },
    {
      "ChangeType": "CreateOffer",
      "ChangeName": "CreateOfferChange",
      "Entity": {
        "Type": "Offer@1.0"
      },
      "DetailsDocument": {
        "ProductId": "$CreateProductChange.Entity.Identifier"
      }
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Name": "New Test Offer",
        "Description": "New offer description"
      }
    },
    {
      "ChangeType": "UpdateLegalTerms",
      "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "Terms": [
          {
            "Type": "LegalTerm",
            "Documents": [
              {
                "Type": "StandardEula",
                "Version": "2022-07-14"
              }
            ]
          }
        ]
      }
    },
    {
      "ChangeType": "UpdateSupportTerms",
```

```

    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Terms": [
        {
          "Type": "SupportTerm",
          "RefundPolicy": "Updated refund policy description"
        }
      ]
    }
  },
  {
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "PricingModel": "Contract",
      "Terms": [
        {
          "Type": "ConfigurableUpfrontPricingTerm",
          "CurrencyCode": "USD",
          "RateCards": [
            {
              "Selector": {
                "Type": "Duration",
                "Value": "P1M"
              },
              "RateCard": [
                {
                  "DimensionKey": "BasicService",
                  "Price": "20"
                },
                {
                  "DimensionKey": "PremiumService",
                  "Price": "25"
                }
              ]
            }
          ],
          "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
          }
        }
      ]
    }
  }
}

```

```

    }
  },
  {
    "Selector": {
      "Type": "Duration",
      "Value": "P12M"
    },
    "RateCard": [
      {
        "DimensionKey": "BasicService",
        "Price": "150"
      },
      {
        "DimensionKey": "PremiumService",
        "Price": "300"
      }
    ],
    "Constraints": {
      "MultipleDimensionSelection": "Allowed",
      "QuantityConfiguration": "Allowed"
    }
  }
]
}
},
{
  "ChangeType": "UpdateRenewalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "RenewalTerm"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {

```

```

        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {}
}
]
}

```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProduct",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "SaaSProduct@1.0"
      },
      "DetailsDocument": {}
    },
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
      },
      "DetailsDocument": {
        "ProductTitle": "Sample product",
        "ShortDescription": "Brief description",
        "LongDescription": "Detailed description",

```



```

        "Highlights": [
            "Sample highlight"
        ],
        "SearchKeywords": [
            "Sample keyword"
        ],
        "Categories": [
            "Data Catalogs"
        ],
        "LogoUrl": "https://bucketname.s3.amazonaws.com/logo.png",
        "VideoUrls": [
            "https://sample.amazonaws.com/awsmvp-video-1"
        ],
        "AdditionalResources": []
    }
},
{
    "ChangeType": "AddDimensions",
    "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": [
        {
            "Key": "BasicService",
            "Description": "Basic Service",
            "Name": "Basic Service",
            "Types": [
                "Entitled"
            ],
            "Unit": "Units"
        },
        {
            "Key": "PremiumService",
            "Description": "Premium Service",
            "Name": "Premium Service",
            "Types": [
                "Entitled"
            ],
            "Unit": "Units"
        }
    ]
},
{

```

```

    "ChangeType": "AddDeliveryOptions",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "DeliveryOptions": [
        {
          "Details": {
            "SaaSUrlDeliveryOptionDetails": {
              "FulfillmentUrl": "https://www.aws.amazon.com/
marketplace/management"
            }
          }
        }
      ]
    }
  },
  {
    "ChangeType": "ReleaseProduct",
    "Entity": {
      "Type": "SaaSProduct@1.0",
      "Identifier": "$CreateProductChange.Entity.Identifier"
    },
    "DetailsDocument": {}
  },
  {
    "ChangeType": "CreateOffer",
    "ChangeName": "CreateOfferChange",
    "Entity": {
      "Type": "Offer@1.0"
    },
    "DetailsDocument": {
      "ProductId": "$CreateProductChange.Entity.Identifier"
    }
  },
  {
    "ChangeType": "UpdateInformation",
    "Entity": {
      "Type": "Offer@1.0",
      "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
      "Name": "New Test Offer",

```

```

        "Description": "New offer description"
    }
},
{
    "ChangeType": "UpdateLegalTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "LegalTerm",
                "Documents": [
                    {
                        "Type": "StandardEula",
                        "Version": "2022-07-14"
                    }
                ]
            }
        ]
    }
},
{
    "ChangeType": "UpdateSupportTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },
    "DetailsDocument": {
        "Terms": [
            {
                "Type": "SupportTerm",
                "RefundPolicy": "Updated refund policy description"
            }
        ]
    }
},
{
    "ChangeType": "UpdatePricingTerms",
    "Entity": {
        "Type": "Offer@1.0",
        "Identifier": "$CreateOfferChange.Entity.Identifier"
    },

```

```
"DetailsDocument": {
  "PricingModel": "Contract",
  "Terms": [
    {
      "Type": "ConfigurableUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P1M"
          },
          "RateCard": [
            {
              "DimensionKey": "BasicService",
              "Price": "20"
            },
            {
              "DimensionKey": "PremiumService",
              "Price": "25"
            }
          ],
          "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
          }
        },
        {
          "Selector": {
            "Type": "Duration",
            "Value": "P12M"
          },
          "RateCard": [
            {
              "DimensionKey": "BasicService",
              "Price": "150"
            },
            {
              "DimensionKey": "PremiumService",
              "Price": "300"
            }
          ],
          "Constraints": {
            "MultipleDimensionSelection": "Allowed",
```

```

        "QuantityConfiguration": "Allowed"
      }
    ]
  }
},
{
  "ChangeType": "UpdateRenewalTerms",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {
    "Terms": [
      {
        "Type": "RenewalTerm"
      }
    ]
  }
},
{
  "ChangeType": "ReleaseOffer",
  "Entity": {
    "Type": "Offer@1.0",
    "Identifier": "$CreateOfferChange.Entity.Identifier"
  },
  "DetailsDocument": {}
}
]
}

```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Publish my SaaS product and associated public offer (product will be in limited
state by default)

```

```
CAPI-05A
"""

import os

import utils.start_changeset as sc
import utils.stringify_details as sd

def main(change_set=None):
    if change_set is None:
        fname = "changeset1.json"
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
        stringified_change_set = sd.stringify_changeset(change_set_file)

    else:
        stringified_change_set = change_set

    response = sc.usage_demo(
        stringified_change_set,
        "publish saas product and associated public offer",
    )

    return response

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

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.


For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Publish a SaaS product and associated public offer from an existing draft using an AWS SDK

The following code example shows how to publish a SaaS product and associated public offer from an existing draft. The product will be in a limited state by default.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateVisibility",
      "ChangeName": "CreateProductChange",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-1111111111111111"
      },
      "DetailsDocument": {
        "TargetVisibility": "Public"
      }
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Update dimensions on an AMI or SaaS product using an AWS SDK

The following code examples show how to update dimensions on an AMI or SaaS product.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.


To run this example, pass the following JSON changeset to `RunChangesets` in *Utilities to start a changeset* from the **Utilities** section.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDimensions",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-111111111111"
      },
      "DetailsDocument": [
        {
          "Key": "BasicService",
          "Types": [
            "Entitled"
          ],
          "Name": "Some new name",
          "Description": "Some new description"
        }
      ]
    }
  ]
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDimensions",
      "Entity": {
        "Type": "SaaSProduct@1.0",
        "Identifier": "prod-111111111111"
      },
      "DetailsDocument": [
        {
          "Key": "BasicService",
          "Types": [
            "Entitled"
          ],
          "Name": "Some new name",
          "Description": "Some new description"
        }
      ]
    }
  ]
}
```

Run this script to start the changeset. Helper functions are defined in *Utilities to start a changeset* from the **Utilities** section.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
```

Shows how to use the AWS SDK for Python (Boto3) to update (e.g name) dimensions on my AMI or SaaS product

```
CAPI-24
```

```
"""
```

```
import os
```

```
import utils.start_changeset as sc
```

```
import utils.stringify_details as sd
```

```
def main(change_set=None):
```

```
    if change_set is None:
```

```
        fname = "changeset.json"
```

```
        change_set_file = os.path.join(os.path.dirname(__file__), fname)
```

```
        stringified_change_set = sd.stringify_changeset(change_set_file)
```

```
    else:
```

```
        stringified_change_set = change_set
```

```
    response = sc.usage_demo(
```

```
        stringified_change_set,
```

```
        "Update name dimensions on my AMI or SaaS product",
```

```
    )
```

```
    return response
```

```
if __name__ == "__main__":
```

```
    main()
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Utilities for AWS Marketplace Catalog API using AWS SDKs

The following code examples show how to use AWS Marketplace Catalog API with AWS SDKs.

Examples

- [Utilities to start a changeset using an AWS SDK](#)

Utilities to start a changeset using an AWS SDK

The following code examples show how to define utilities to start a changeset.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

Utility to load a changeset from a JSON file and start processing it.

```
package com.example.awsmarketplace.catalogapi;

import java.io.ByteArrayInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;

import org.apache.commons.io.IOUtils;
import org.apache.commons.lang3.StringUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.core.document.Document;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.protocols.json.internal.unmarshall.document.DocumentUnmarshaller;
import software.amazon.awssdk.protocols.jsoncore.JsonNodeParser;
import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import software.amazon.awssdk.services.marketplacecatalog.model.Change;
import software.amazon.awssdk.services.marketplacecatalog.model.Entity;
```

```
import
    software.amazon.awssdk.services.marketplacecatalog.model.StartChangeSetRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.StartChangeSetResponse;
import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.ToNumberPolicy;
import com.example.awsmarketplace.catalogapi.Entity.ChangeSet;
import com.example.awsmarketplace.catalogapi.Entity.ChangeSetEntity;
import com.example.awsmarketplace.catalogapi.Entity.Root;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;
import com.example.awsmarketplace.utils.StringSerializer;

/**
 * Before running this Java V2 code example, convert all Details attribute to
 * DetailsDocument if any
 */

public class RunChangesets {

    private static final Gson GSON = new GsonBuilder()
        .setObjectToNumberStrategy(ToNumberPolicy.LAZILY_PARSED_NUMBER)
        .registerTypeAdapter(String.class, new StringSerializer())
        .create();

    public static void main(String[] args) {

        // input json can be specified here or passed from input parameter
        String inputChangeSetFile = "changeSets/offers/
CreateReplacementOfferFromAGWithContractPricingDetailDocument.json";

        if (args.length > 0)
            inputChangeSetFile = args[0];

        // parse the input changeset file to string for process
        String changeSetsInput = readChangeSetToString(inputChangeSetFile);

        // process the changeset request
        try {
            StartChangeSetResponse result = getChangeSetRequestResult(changeSetsInput);
            ReferenceCodesUtils.formatOutput(result);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
}

public static StartChangeSetResponse getChangeSetRequestResult(String
changeSetsInput) throws IOException {

    //set up AWS credentials
    MarketplaceCatalogClient marketplaceCatalogClient =
        MarketplaceCatalogClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    //changeset list to save all the changesets in the changesets file
    List<Change> changeSetLists = new ArrayList<Change>();

    // read all changesets into object
    Root root = GSON.fromJson(changeSetsInput, Root.class);

    // process each changeset and add each changeset request to changesets list
    for (ChangeSet cs : root.changeSet) {

        ChangeSetEntity entity = cs.Entity;
        String entityType = entity.Type;
        String entityIdIdentifier = StringUtils.defaultIfBlank(entity.Identifier, null);
        Document detailsDocument = getDocumentFromObject(cs.DetailsDocument);

        Entity awsEntity =
            Entity.builder()
                .type(entityType)
                .identifier(entityIdentifier)
                .build();

        Change inputChangeRequest =
            Change.builder()
                .changeType(cs.ChangeType)
                .changeName(cs.ChangeName)
                .entity(awsEntity)
                .detailsDocument(detailsDocument)
                .build();

        changeSetLists.add(inputChangeRequest);
    }

    // process all changeset requests
```

```
StartChangeSetRequest startChangeSetRequest =
    StartChangeSetRequest.builder()
        .catalog(root.catalog)
        .changeSet(changeSetLists)
        .build();

StartChangeSetResponse result =
marketplaceCatalogClient.startChangeSet(startChangeSetRequest);

return result;
}

public static Document getDocumentFromObject(Object detailsObject) {

    String detailsString = "{}";
    try {
        detailsString = IOUtils.toString(new
ByteArrayInputStream(GSON.toJson(detailsObject).getBytes()), "UTF-8");
    } catch (IOException e) {
        e.printStackTrace();
    }

    JsonNodeParser jsonNodeParser = JsonNodeParser.create();
    Document doc = jsonNodeParser.parse(detailsString).visit(new
DocumentUnmarshaller());
    return doc;
}

public static String readChangeSetToString (String inputChangeSetFile) {

    InputStream changesetInputStream =
RunChangesets.class.getClassLoader().getResourceAsStream(inputChangeSetFile);

    String changeSetsInput = null;

    try {
        changeSetsInput = IOUtils.toString(changesetInputStream, "UTF-8");
    } catch (IOException e) {
        e.printStackTrace();
    }

    return changeSetsInput;
}
```

```
}  
}
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

Utility to start a changeset.

```
"""  
Purpose:  
  
Generic function to start a changeset  
"""  
  
import logging  
  
import boto3  
from botocore.exceptions import ClientError  
  
logger = logging.getLogger(__name__)  
  
def generate_changeset(mp_client, change_set, change_set_name):  
    """  
    Start changeset  
    """  
    try:  
        response = mp_client.start_change_set(  
            Catalog="AWSMarketplace",  
            ChangeSet=change_set,  
            ChangeSetName=change_set_name,  
        )
```

```
        logger.info("Changeset created!")
        logger.info("ChangeSet ID: %s", response["ChangeSetId"])
        logger.info("ChangeSet ARN: %s", response["ChangeSetArn"])

        return response

    except ClientError as e:
        logger.exception("Unexpected error: %s", e)
        raise

def usage_demo(change_set, change_set_name):
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Executing changeset: " + change_set_name)
    print("-" * 88)

    mp_client = boto3.client("marketplace-catalog")

    response = generate_changeset(mp_client, change_set, change_set_name)

    return response

    print("-" * 88)
```

Utility to load a changeset from a JSON file.

```
"""
Purpose:

This module will stringify the details sections of a changeset file.
"""

import json

def pretty_print(response):
    json_object = json.dumps(response, indent=4)
    print(json_object)
```



```
# open json file from path
def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

def stringify_details_sections(json_object):
    """
    Loops through every change type in the changeset to look for non-empty
    details section and stringifies them
    """
    for change_type in json_object["ChangeSet"]:
        # Only stringify details section if it is not empty
        if "Details" in change_type and change_type["Details"] != "{}":
            string_details = json.dumps(change_type["Details"])
            change_type["Details"] = string_details
        else:
            pass

    return json_object["ChangeSet"]

def stringify_changeset(file_path):
    changeset_file = open_json_file(file_path)
    changeset_stringified = stringify_details_sections(changeset_file)

    return changeset_stringified
```

- For API details, see [StartChangeSet](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Code examples for AWS Marketplace Agreement API using AWS SDKs

The following code examples show how to use AWS Marketplace Agreement API with an AWS software development kit (SDK).

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Code examples

- [Agreements for AWS Marketplace Agreement API using AWS SDKs](#)
 - [Get all agreement IDs using an AWS SDK](#)
 - [Get all agreements using an AWS SDK](#)
 - [Get customer ID from an agreement using an AWS SDK](#)
 - [Get financial details from an agreement using an AWS SDK](#)
 - [Get free trial details from an agreement using an AWS SDK](#)
 - [Get information about an agreement using an AWS SDK](#)
 - [Get product and offer details from an agreement using an AWS SDK](#)
 - [Get the EULA of an agreement using an AWS SDK](#)
 - [Get the auto renewal terms of an agreement using an AWS SDK](#)
 - [Get the dimensions purchased in an agreement using an AWS SDK](#)
 - [Get the instances of each dimension purchased in an agreement using an AWS SDK](#)
 - [Get the payment schedule of an agreement using an AWS SDK](#)
 - [Get the pricing per dimension in an agreement using an AWS SDK](#)
 - [Get the pricing type of an agreement using an AWS SDK](#)
 - [Get the product type of an agreement using an AWS SDK](#)
 - [Get the status of an agreement using an AWS SDK](#)
 - [Get the support terms of an agreement using an AWS SDK](#)
 - [Get the terms of an agreement using an AWS SDK](#)
 - [Search for agreements by account ID using an AWS SDK](#)
 - [Search for agreements by agreement ID using an AWS SDK](#)
 - [Search for agreements by end date using an AWS SDK](#)
 - [Search for agreements by offer ID using an AWS SDK](#)
 - [Search for agreements by product ID using an AWS SDK](#)
 - [Search for agreements by status using an AWS SDK](#)
- [Search for agreements with one custom filter using an AWS SDK](#)

- [Search for agreements with two custom filters using an AWS SDK](#)

Agreements for AWS Marketplace Agreement API using AWS SDKs

The following code examples show how to use AWS Marketplace Agreement API with AWS SDKs.

Examples

- [Get all agreement IDs using an AWS SDK](#)
- [Get all agreements using an AWS SDK](#)
- [Get customer ID from an agreement using an AWS SDK](#)
- [Get financial details from an agreement using an AWS SDK](#)
- [Get free trial details from an agreement using an AWS SDK](#)
- [Get information about an agreement using an AWS SDK](#)
- [Get product and offer details from an agreement using an AWS SDK](#)
- [Get the EULA of an agreement using an AWS SDK](#)
- [Get the auto renewal terms of an agreement using an AWS SDK](#)
- [Get the dimensions purchased in an agreement using an AWS SDK](#)
- [Get the instances of each dimension purchased in an agreement using an AWS SDK](#)
- [Get the payment schedule of an agreement using an AWS SDK](#)
- [Get the pricing per dimension in an agreement using an AWS SDK](#)
- [Get the pricing type of an agreement using an AWS SDK](#)
- [Get the product type of an agreement using an AWS SDK](#)
- [Get the status of an agreement using an AWS SDK](#)
- [Get the support terms of an agreement using an AWS SDK](#)
- [Get the terms of an agreement using an AWS SDK](#)
- [Search for agreements by account ID using an AWS SDK](#)
- [Search for agreements by agreement ID using an AWS SDK](#)
- [Search for agreements by end date using an AWS SDK](#)
- [Search for agreements by offer ID using an AWS SDK](#)
- [Search for agreements by product ID using an AWS SDK](#)
- [Search for agreements by status using an AWS SDK](#)

- [Search for agreements with one custom filter using an AWS SDK](#)
- [Search for agreements with two custom filters using an AWS SDK](#)

Get all agreement IDs using an AWS SDK

The following code examples show how to get all agreement IDs.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAllAgreementsIds {
```

```
/*
 * Get all purchase agreements ids with party type = proposer;
 * Depend on the number of agreements in your account, this code may take some
 time to finish.
 */
public static void main(String[] args) {

    List<String> agreementIds = getAllAgreementIds();

    ReferenceCodesUtils.formatOutput(agreementIds);

}

public static List<String> getAllAgreementIds() {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    // get all filters
    Filter partyType = Filter.builder().name(PARTY_TYPE_FILTER_NAME)
        .values(PARTY_TYPE_FILTER_VALUE_PROPOSER).build();

    Filter agreementType = Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME)
        .values(AGREEMENT_TYPE_FILTER_VALUE_PURCHASEAGREEMENT).build();

    List<Filter> searchFilters = new ArrayList<Filter>();

    searchFilters.addAll(Arrays.asList(partyType, agreementType));

    // Save all results in a list array
    List<AgreementViewSummary> agreementSummaryList = new
    ArrayList<AgreementViewSummary>();

    SearchAgreementsRequest searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(searchFilters)
            .build();

    SearchAgreementsResponse searchAgreementsResponse =
    marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);
}
```

```
agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());

while (searchAgreementsResponse.nextToken() != null &&
searchAgreementsResponse.nextToken().length() > 0) {
    searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .nextToken(searchAgreementsResponse.nextToken())
            .filters(searchFilters)
            .build();
    searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());
}

List<String> agreementIds = new ArrayList<String>();
for (AgreementViewSummary summary : agreementSummaryList) {
    agreementIds.add(summary.agreementId());
}
return agreementIds;
}
}
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
```

Purpose

Shows how to use the AWS SDK for Python (Boto3) to get all agreement ids

AG-09

```
"""
```

```
import logging
```

```
import boto3
```

```
from botocore.exceptions import ClientError
```

```
mp_client = boto3.client("marketplace-agreement")
```

```
logger = logging.getLogger(__name__)
```

```
MAX_PAGE_RESULTS = 10
```

```
def get_agreements():
```

```
    AgreementSummaryList = []
```

```
    agreement_id_list = []
```

```
    try:
```

```
        agreements = mp_client.search_agreements(  
            catalog="AWSMarketplace",
```

```
            maxResults=MAX_PAGE_RESULTS,
```

```
            filters=[
```

```
                {"name": "PartyType", "values": ["Proposer"]},
```

```
                {"name": "AgreementType", "values": ["PurchaseAgreement"]},
```

```
            ],
```

```
        )
```

```
    except ClientError as e:
```

```
        logger.error("Could not complete search_agreements request.")
```

```
        raise
```

```
    AgreementSummaryList.extend(agreements["agreementViewSummaries"])
```

```
    while "nextToken" in agreements and agreements["nextToken"] is not None:
```

```
        try:
```

```
            agreements = mp_client.search_agreements(  
                catalog="AWSMarketplace",
```

```
                maxResults=MAX_PAGE_RESULTS,
```

```
                nextToken=agreements["nextToken"],
```

```
                filters=[
```

```
                    {"name": "PartyType", "values": ["Proposer"]},
```

```
        {"name": "AgreementType", "values": ["PurchaseAgreement"]},
    ],
)
except ClientError as e:
    logger.error("Could not complete search_agreements request.")
    raise

AgreementSummaryList.extend(agreements["agreementViewSummaries"])

for agreement in AgreementSummaryList:
    agreement_id_list.append(agreement["agreementId"])

return agreement_id_list

if __name__ == "__main__":
    agreement_id_list = get_agreements()

    print(agreement_id_list)
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get all agreements using an AWS SDK

The following code examples show how to get all agreements.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.


```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAllAgreements {

    /**
     * Get all purchase agreements with party type = proposer;
     * Depend on the number of agreements in your account, this code may take some
     * time to finish.
     */
    public static void main(String[] args) {

        List<AgreementViewSummary> agreementSummaryList = getAllAgreements();

        ReferenceCodesUtils.formatOutput(agreementSummaryList);
    }

    public static List<AgreementViewSummary> getAllAgreements() {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
    }
}
```

```
.build();

// get all filters

Filter partyType = Filter.builder().name(PARTY_TYPE_FILTER_NAME)
    .values(PARTY_TYPE_FILTER_VALUE_PROPOSER).build();

Filter agreementType = Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME)
    .values(AGREEMENT_TYPE_FILTER_VALUE_PURCHASEAGREEMENT).build();

List<Filter> searchFilters = new ArrayList<Filter>();

searchFilters.addAll(Arrays.asList(partyType, agreementType));

// Save all results in a list array

List<AgreementViewSummary> agreementSummaryList = new
ArrayList<AgreementViewSummary>();

SearchAgreementsRequest searchAgreementsRequest =
    SearchAgreementsRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .filters(searchFilters)
        .build();

SearchAgreementsResponse searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());

while (searchAgreementsResponse.nextToken() != null &&
searchAgreementsResponse.nextToken().length() > 0) {
    searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .nextToken(searchAgreementsResponse.nextToken())
            .filters(searchFilters).build();
    searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

    agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());
}
return agreementSummaryList;
}
```

```
}
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get all agreements
AG-01
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")

logger = logging.getLogger(__name__)

MAX_PAGE_RESULTS = 10

party_type_list = ["Proposer"]
agreement_type_list = ["PurchaseAgreement"]

filter_list = [
    {"name": "PartyType", "values": party_type_list},
    {"name": "AgreementType", "values": agreement_type_list},
```

```
]

agreement_results_list = []

def get_agreements(filter_list=filter_list):
    try:
        agreements = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            filters=filter_list,
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise e

    agreement_results_list.extend(agreements["agreementViewSummaries"])

    while "nextToken" in agreements and agreements["nextToken"] is not None:
        try:
            agreements = mp_client.search_agreements(
                catalog="AWSMarketplace",
                maxResults=MAX_PAGE_RESULTS,
                nextToken=agreements["nextToken"],
                filters=filter_list,
            )
        except ClientError as e:
            logger.error("Could not complete search_agreements request.")
            raise e

        agreement_results_list.extend(agreements["agreementViewSummaries"])

    return agreement_results_list

if __name__ == "__main__":
    agreements_list = get_agreements(filter_list)
    helper.pretty_print_datetime(agreements_list)
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get customer ID from an agreement using an AWS SDK

The following code examples show how to get customer ID from an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;

public class GetAgreementCustomerInfo {

    /*
     * Obtain metadata about the customer who created the agreement, such as the
     * customer's AWS Account ID
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;
```

```
DescribeAgreementResponse describeAgreementResponse =
getDescribeAgreementResponse(agreementId);

System.out.println("Customer's AWS Account ID is " +
describeAgreementResponse.acceptor().accountId());

}

public static DescribeAgreementResponse getDescribeAgreementResponse(String
agreementId) {
MarketplaceAgreementClient marketplaceAgreementClient =
MarketplaceAgreementClient.builder()
.httpClient(ApacheHttpClient.builder().build())
.credentialsProvider(ProfileCredentialsProvider.create())
.build();

DescribeAgreementRequest describeAgreementRequest =
DescribeAgreementRequest.builder()
.agreementId(agreementId)
.build();

DescribeAgreementResponse describeAgreementResponse =
marketplaceAgreementClient.describeAgreement(describeAgreementRequest);
return describeAgreementResponse;
}
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get customer AWS account id
from a given agreement
AG-08
"""

import argparse
import logging

import boto3
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")

logger = logging.getLogger(__name__)

def get_agreement_information(agreement_id):
    try:
        response = mp_client.describe_agreement(agreementId=agreement_id)
    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)
            raise e
        else:
            logger.error("Unexpected error: %s", e)
            raise e

    return response

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement-id",
        "-aid",
        help="Provide agreement ID to describe agreement status",
        required=True,
    )
    args = parser.parse_args()
```

```
response = get_agreement_information(agreement_id=args.agreement_id)

print(f"Customer account: {response['acceptor']['accountId']}")
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get financial details from an agreement using an AWS SDK

The following code examples show how to get financial details from an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;
```



```
public class GetAgreementFinancialDetails {

    /**
     * Obtain financial details, such as Total Contract Value of the agreement from
     * a given agreement
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        String totalContractValue = getTotalContractValue(agreementId);

        System.out.println("Total Contract Value is " + totalContractValue);

    }

    public static String getTotalContractValue(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        DescribeAgreementRequest describeAgreementRequest =
            DescribeAgreementRequest.builder()
                .agreementId(agreementId)
                .build();

        DescribeAgreementResponse describeAgreementResponse =
            marketplaceAgreementClient.describeAgreement(describeAgreementRequest);

        String totalContractValue = "N/A";

        if ( describeAgreementResponse.estimatedCharges() != null ) {
            totalContractValue =
                describeAgreementResponse.estimatedCharges().agreementValue()
                    + " "
                    + describeAgreementResponse.estimatedCharges().currencyCode();
        }
        return totalContractValue;
    }
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain financial details, such as Total Contract Value of the agreement from a
given agreement
AG-14

Example Usage: python3 get_agreement_financial_details.py --agreement-id
<agreement-id>
"""

import argparse
import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

mp_client = boto3.client("marketplace-agreement")

def get_agreement_information(agreement_id):
    try:
        agreement = mp_client.describe_agreement(agreementId=agreement_id)

        return agreement

    except ClientError as e:
```

```
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)
        else:
            logger.error("Unexpected error: %s", e)

    return None

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement-id",
        "-aid",
        help="Provide agreement ID to describe agreement status",
        required=True,
    )
    args = parser.parse_args()

    agreement = get_agreement_information(args.agreement_id)

    if agreement is not None:
        print(f"Agreement Id: {args.agreement_id}")
        print(
            f"Agreement Value: {agreement['estimatedCharges']['currencyCode']}
            {agreement['estimatedCharges']['agreementValue']}"
        )

    else:
        print(f"Agreement with ID {args.agreement_id} is not found")
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get free trial details from an agreement using an AWS SDK

The following code examples show how to get free trial details from an agreement.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.FreeTrialPricingTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;

import java.util.ArrayList;
import java.util.List;

import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsFreeTrialDetails {

    /**
     * Obtain the details from an agreement of a free trial I have provided to the
     customer
     */
    public static void main(String[] args) {
```

```
String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

List<FreeTrialPricingTerm> freeTrialPricingTerms =
getFreeTrialPricingTerms(agreementId);

ReferenceCodesUtils.formatOutput(freeTrialPricingTerms);
}

public static List<FreeTrialPricingTerm> getFreeTrialPricingTerms(String
agreementId) {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    GetAgreementTermsRequest getAgreementTermsRequest =
        GetAgreementTermsRequest.builder().agreementId(agreementId)
            .build();

    GetAgreementTermsResponse getAgreementTermsResponse =
        marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);

    List<FreeTrialPricingTerm> freeTrialPricingTerms = new
        ArrayList<FreeTrialPricingTerm>();

    for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
        if (acceptedTerm.freeTrialPricingTerm() != null) {
            freeTrialPricingTerms.add(acceptedTerm.freeTrialPricingTerm());
        }
    }
    return freeTrialPricingTerms;
}
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the details from an agreement of a free trial I have provided to the
customer
AG-20

Example Usage: python3 get_agreement_free_trial_details.py --agreement-id
<agreement-id>
"""

import argparse
import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

mp_client = boto3.client("marketplace-agreement")

def get_agreement_terms(agreement_id):
    try:
        agreement = mp_client.get_agreement_terms(agreementId=agreement_id)
        return agreement

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)
```

```
        else:
            logger.error("Unexpected error: %s", e)

    return None

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement-id",
        "-aid",
        help="Provide agreement ID to describe agreement status",
        required=True,
    )
    args = parser.parse_args()

    agreement = get_agreement_terms(agreement_id=args.agreement_id)

    if agreement is not None:
        freetrial_found = False

        for term in agreement["acceptedTerms"]:
            if "freeTrialPricingTerm" in term.keys():
                helper.pretty_print_datetime(term)
                freetrial_found = True

    if not freetrial_found:
        print(f"No free trial term found for agreement: {args.agreement_id}")
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get information about an agreement using an AWS SDK

The following code examples show how to get information about an agreement.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;

public class DescribeAgreement {

    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        DescribeAgreementResponse describeAgreementResponse = getResponse(agreementId);

        ReferenceCodesUtils.formatOutput(describeAgreementResponse);

    }

    public static DescribeAgreementResponse getResponse(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
```



```
.credentialsProvider(ProfileCredentialsProvider.create())
    .build();

DescribeAgreementRequest describeAgreementRequest =
    DescribeAgreementRequest.builder()
        .agreementId(agreementId)
        .build();

DescribeAgreementResponse describeAgreementResponse =
    marketplaceAgreementClient.describeAgreement(describeAgreementRequest);
return describeAgreementResponse;
}

}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get agreement information
AG-07
"""

import argparse
import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError
```

```
mp_client = boto3.client("marketplace-agreement")

logger = logging.getLogger(__name__)

def get_agreement_information(agreement_id):
    try:
        response = mp_client.describe_agreement(agreementId=agreement_id)
    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)
            raise e
        else:
            logger.error("Unexpected error: %s", e)
            raise e

    return response

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement-id",
        "-aid",
        help="Provide agreement ID to describe agreement status",
        required=True,
    )
    args = parser.parse_args()

    response = get_agreement_information(agreement_id=args.agreement_id)

    helper.pretty_print_datetime(response)
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get product and offer details from an agreement using an AWS SDK

The following code examples show how to get product and offer details from an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;
import software.amazon.awssdk.services.marketplaceagreement.model.Resource;

import java.util.ArrayList;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;

public class GetProductAndOfferDetailFromAgreement {
```

```
public static void main(String[] args) {

    // call Agreement API to get offer and product information for the agreement

    String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

    List<DescribeEntityResponse> entityResponseList = getEntities(agreementId);

    for (DescribeEntityResponse response : entityResponseList) {
        ReferenceCodesUtils.formatOutput(response);
    }
}

public static List<DescribeEntityResponse> getEntities(String agreementId) {
    List<DescribeEntityResponse> entityResponseList = new
    ArrayList<DescribeEntityResponse> ();

    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    DescribeAgreementRequest describeAgreementRequest =
        DescribeAgreementRequest.builder()
            .agreementId(agreementId)
            .build();

    DescribeAgreementResponse describeAgreementResponse =
    marketplaceAgreementClient.describeAgreement(describeAgreementRequest);

    // get offer id for the given agreement

    String offerId = describeAgreementResponse.proposalSummary().offerId();

    // get all the product ids for this agreement

    List<String> productIds = new ArrayList<String>();
    for (Resource resource :
    describeAgreementResponse.proposalSummary().resources()) {
        productIds.add(resource.id());
    }
}
```

```
// call Catalog API to get the details of the offer and products

MarketplaceCatalogClient marketplaceCatalogClient =
    MarketplaceCatalogClient.builder()
        .httpClient(ApacheHttpClient.builder().build())
        .credentialsProvider(ProfileCredentialsProvider.create())
        .build();

DescribeEntityRequest describeEntityRequest =
    DescribeEntityRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .entityId(offerId).build();

DescribeEntityResponse describeEntityResponse =
marketplaceCatalogClient.describeEntity(describeEntityRequest);

entityResponseList.add(describeEntityResponse);

for (String productId : productIds) {
    describeEntityRequest =
        DescribeEntityRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityId(productId).build();
    describeEntityResponse =
marketplaceCatalogClient.describeEntity(describeEntityRequest);
    System.out.println("Print details for product " + productId);
    entityResponseList.add(describeEntityResponse);
}
return entityResponseList;
}
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get product and offer details
in a given agreement
AG-10
"""

import argparse
import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

mpa_client = boto3.client("marketplace-agreement")
mpc_client = boto3.client("marketplace-catalog")

logger = logging.getLogger(__name__)

def get_agreement_information(agreement_id):
    """
    Returns information about a given agreement
    Args: agreement_id str: Entity to return
    Returns: dict: Dictionary of agreement information
    """

    try:
        agreement = mpa_client.describe_agreement(agreementId=agreement_id)
```

```
        return agreement

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)
        else:
            logger.error("Unexpected error: %s", e)

def get_entity_information(entity_id):
    """
    Returns information about a given entity
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of entity information
    """

    try:
        response = mpc_client.describe_entity(
            Catalog="AWSMarketplace",
            EntityId=entity_id,
        )

        return response

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Entity with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def get_agreement_components(agreement_id):
    agreement_component_list = []

    agreement = get_agreement_information(agreement_id)

    if agreement is not None:
        productIds = []
        for resource in agreement["proposalSummary"]["resources"]:
            productIds.append(resource["id"])

        for product_id in productIds:
            product_document = get_entity_information(product_id)
```

```
        product_document_dict = {}
        product_document_dict["product_id"] = product_id
        product_document_dict["document"] = product_document
        agreement_component_list.append(product_document_dict)

    offerId = agreement["proposalSummary"]["offerId"]

    offer_document = get_entity_information(offerId)

    offer_document_dict = {}
    offer_document_dict["offer_id"] = offerId
    offer_document_dict["document"] = offer_document
    agreement_component_list.append(offer_document_dict)

    return agreement_component_list

else:
    print("Agreement with ID " + args.agreement_id + " is not found")

if __name__ == "__main__":
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement_id",
        "-aid",
        help="Provide agreement ID to search for product and offer detail",
        required=True,
    )
    args = parser.parse_args()

    product_offer_detail =
    get_agreement_components(agreement_id=args.agreement_id)

    helper.pretty_print_datetime(product_offer_detail)
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the EULA of an agreement using an AWS SDK

The following code examples show how to get the EULA of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.DocumentItem;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

import java.util.ArrayList;
import java.util.List;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsEula {

    /*
```

```
    * Obtain the EULA I have entered into with my customer via the agreement
    */
public static void main(String[] args) {

    String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

    List<DocumentItem> legalEulaArray = getLegalEula(agreementId);

    ReferenceCodesUtils.formatOutput(legalEulaArray);
}

public static List<DocumentItem> getLegalEula(String agreementId) {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    GetAgreementTermsRequest getAgreementTermsRequest =
        GetAgreementTermsRequest.builder().agreementId(agreementId)
            .build();

    GetAgreementTermsResponse getAgreementTermsResponse =
        marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);

    List<DocumentItem> legalEulaArray = new ArrayList<>();

    getAgreementTermsResponse.acceptedTerms().stream()
        .filter(acceptedTerm -> acceptedTerm.legalTerm() != null &&
            acceptedTerm.legalTerm().hasDocuments())
        .flatMap(acceptedTerm -> acceptedTerm.legalTerm().documents().stream())
        .filter(docItem -> docItem.type() != null)
        .forEach(legalEulaArray::add);
    return legalEulaArray;
}
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the EULA I have entered into with my customer via the agreement
AG-18
"""

import json
import logging
import os

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

# agreement id
AGREEMENT_ID = "agmt-11111111111111111111111111111111"

# to use sample file or not
USE_SAMPLE_FILE = False
SAMPLE_FILE_NAME = "mockup_agreement_terms.json"

# attribute name
ROOT_ELEM = "acceptedTerms"
TERM_NAME = "legalTerm"
CONFIG_ELEM = "configuration"
ATTRIBUTE_NAME = "documents"
```

```
def get_agreement_information(mp_client, entity_id):
    """
    Returns customer AWS Account id about a given agreement
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of agreement information
    """

    try:
        if USE_SAMPLE_FILE:
            sample_file = os.path.join(os.path.dirname(__file__),
SAMPLE_FILE_NAME)
            terms = open_json_file(sample_file)
        else:
            terms = mp_client.get_agreement_terms(agreementId=entity_id)

        legalEulaArray = []
        for term in terms[ROOT_ELEM]:
            if TERM_NAME in term and ATTRIBUTE_NAME in term[TERM_NAME]:
                docs = term[TERM_NAME][ATTRIBUTE_NAME]
                for doc in docs:
                    if "type" in doc:
                        legalEulaArray.append(doc)
        return legalEulaArray

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

    helper.pretty_print_datetime(get_agreement_information(mp_client,
AGREEMENT_ID))

    # open json file from path
```

```
def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the auto renewal terms of an agreement using an AWS SDK

The following code examples show how to get the auto renewal terms of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
```

```
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

public class GetAgreementAutoRenewal {

    /*
     * Obtain the auto-renewal status of the agreement
     */

    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        String autoRenewal = getAutoRenewal(agreementId);

        System.out.println("Auto-Renewal status is " + autoRenewal);
    }

    public static String getAutoRenewal(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
            GetAgreementTermsRequest.builder()
                .agreementId(agreementId)
                .build();

        GetAgreementTermsResponse getAgreementTermsResponse =
            marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);

        String autoRenewal = "No Auto Renewal";

        for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
            if (acceptedTerm.renewalTerm() != null &&
                acceptedTerm.renewalTerm().configuration() != null
                    && acceptedTerm.renewalTerm().configuration().enableAutoRenew() != null) {
                autoRenewal =
                    String.valueOf(acceptedTerm.renewalTerm().configuration().enableAutoRenew().booleanValue());
            }
        }
    }
}
```



```
# to use sample file or not
USE_SAMPLE_FILE = False
SAMPLE_FILE_NAME = "mockup_agreement_terms.json"

# attribute name
ROOT_ELEM = "acceptedTerms"
TERM_NAME = "renewalTerm"
CONFIG_ELEM = "configuration"
ATTRIBUTE_NAME = "enableAutoRenew"

def get_agreement_information(mp_client, entity_id):
    """
    Returns customer AWS Account id about a given agreement
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of agreement information
    """

    try:
        if USE_SAMPLE_FILE:
            sample_file = os.path.join(os.path.dirname(__file__),
SAMPLE_FILE_NAME)
            terms = open_json_file(sample_file)
        else:
            terms = mp_client.get_agreement_terms(agreementId=entity_id)

        auto_renewal = "No Auto Renewal"
        for term in terms[ROOT_ELEM]:
            if TERM_NAME in term:
                if CONFIG_ELEM in term[TERM_NAME]:
                    auto_renewal = term[TERM_NAME][CONFIG_ELEM][ATTRIBUTE_NAME]
                    break
        return auto_renewal

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")
```



```
print("-" * 88)
print("Looking for an agreement in the AWS Marketplace.")
print("-" * 88)

mp_client = boto3.client("marketplace-agreement")

agreement = get_agreement_information(mp_client, AGREEMENT_ID)

if agreement is not None:
    print(f"Auto Renewal is {agreement}")
else:
    print("Agreement with ID " + AGREEMENT_ID + " is not found")

# open json file from path
def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the dimensions purchased in an agreement using an AWS SDK

The following code examples show how to get the dimensions purchased in an agreement.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import software.amazon.awssdk.services.marketplaceagreement.model.Dimension;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

import java.util.ArrayList;
import java.util.List;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsDimensionPurchased {

    /*
     * Obtain the dimensions the buyer has purchased from me via the agreement
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        List<String> dimensionKeys = getDimensionKeys(agreementId);
```

```
ReferenceCodesUtils.formatOutput(dimensionKeys);
}

public static List<String> getDimensionKeys(String agreementId) {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    GetAgreementTermsRequest getAgreementTermsRequest =
        GetAgreementTermsRequest.builder().agreementId(agreementId)
            .build();


    GetAgreementTermsResponse getAgreementTermsResponse =
        marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);

    List<String> dimensionKeys = new ArrayList<String>();
    for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
        if (acceptedTerm.configurableUpfrontPricingTerm() != null) {
            if
            (acceptedTerm.configurableUpfrontPricingTerm().configuration().selectorValue() !=
            null) {
                List<Dimension> dimensions =
                    acceptedTerm.configurableUpfrontPricingTerm().configuration().dimensions();
                for (Dimension dimension : dimensions) {
                    dimensionKeys.add(dimension.dimensionKey());
                }
            }
        }
    }
    return dimensionKeys;
}
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the dimensions the buyer has purchased from me via the agreement
AG-28
"""

import json
import logging
import os

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

# agreement id
AGREEMENT_ID = "agmt-11111111111111111111111111111111"

# to use sample file or not
USE_SAMPLE_FILE = False
SAMPLE_FILE_NAME = "mockup_agreement_terms.json"

# attribute name
ROOT_ELEM = "acceptedTerms"
TERM_NAME = "configurableUpfrontPricingTerm"
CONFIG_ELEM = "configuration"
ATTRIBUTE_NAME = "selectorValue"
```

```
def get_agreement_information(mp_client, entity_id):
    """
    Returns customer AWS Account id about a given agreement
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of agreement information
    """

    try:
        if USE_SAMPLE_FILE:
            sample_file = os.path.join(os.path.dirname(__file__),
SAMPLE_FILE_NAME)
            terms = open_json_file(sample_file)
        else:
            terms = mp_client.get_agreement_terms(agreementId=entity_id)

        dimensionKeys = []

        for term in terms[ROOT_ELEM]:
            if TERM_NAME in term:
                if CONFIG_ELEM in term[TERM_NAME]:
                    confParam = term[TERM_NAME][CONFIG_ELEM]
                    if ATTRIBUTE_NAME in confParam:
                        if "dimensions" in confParam:
                            for dimension in confParam["dimensions"]:
                                if "dimensionKey" in dimension:
                                    dimensionKey = dimension["dimensionKey"]
                                    print(f"Dimension Key: {dimensionKey}")
                                    dimensionKeys.append(dimensionKey)

        return dimensionKeys

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", entity_id)
        else:
            logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace.")
    print("-" * 88)
```

```
mp_client = boto3.client("marketplace-agreement")

helper.pretty_print_datetime(get_agreement_information(mp_client,
AGREEMENT_ID))

# open json file from path

def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the instances of each dimension purchased in an agreement using an AWS SDK

The following code examples show how to get the instances of each dimension purchased in an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
```

```
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import software.amazon.awssdk.services.marketplaceagreement.model.Dimension;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsDimensionInstances {

    /**
     * get instances of each dimension that buyer has purchased in the agreement
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        Map<String, List<Dimension>> dimensionMap = getDimensions(agreementId);

        ReferenceCodesUtils.formatOutput(dimensionMap);
    }

    public static Map<String, List<Dimension>> getDimensions(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
```

```
GetAgreementTermsRequest.builder().agreementId(agreementId)
    .build();

GetAgreementTermsResponse getAgreementTermsResponse =
marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);


Map<String, List<Dimension>> dimensionMap = new HashMap<String,
List<Dimension>>();

for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
    List<Dimension> dimensionsList = new ArrayList<Dimension>();
    if (acceptedTerm.configurableUpfrontPricingTerm() != null) {
        String selectorValue = "";
        if (acceptedTerm.configurableUpfrontPricingTerm().configuration() != null) {
            if
            (acceptedTerm.configurableUpfrontPricingTerm().configuration().selectorValue() !=
            null) {
                selectorValue =
                acceptedTerm.configurableUpfrontPricingTerm().configuration().selectorValue();
            }
            if
            (acceptedTerm.configurableUpfrontPricingTerm().configuration().hasDimensions())
            {
                dimensionsList =
                acceptedTerm.configurableUpfrontPricingTerm().configuration().dimensions();
            }
        }
        if (selectorValue.length() > 0) {
            dimensionMap.put(selectorValue, dimensionsList);
        }
    }
}
return dimensionMap;
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain instances of each dimension that buyer has purchased in the agreement
AG-30
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

# agreement id
AGREEMENT_ID = "agmt-11111111111111111111111111111111"

# attribute name
ROOT_ELEM = "acceptedTerms"
TERM_NAME = "configurableUpfrontPricingTerm"
CONFIG_ELEM = "configuration"
ATTRIBUTE_NAME = "selectorValue"

logger = logging.getLogger(__name__)

def get_agreement_information(mp_client, entity_id):
    """
    Returns customer AWS Account id about a given agreement
    Args: entity_id str: Entity to return
    """
```

```

Returns: dict: Dictionary of agreement information
"""

try:
    terms = mp_client.get_agreement_terms(agreementId=entity_id)
    dimensionKeyValueMap = {}
    for term in terms[ROOT_ELEM]:
        if TERM_NAME in term:
            if CONFIG_ELEM in term[TERM_NAME]:
                confParam = term[TERM_NAME][CONFIG_ELEM]
                if ATTRIBUTE_NAME in confParam:
                    selectValue = confParam["selectorValue"]
                    dimensionKeyValueMap["selectorValue"] = selectValue
                    if "dimensions" in confParam:
                        dimensionKeyValueMap["dimensions"] =
confParam["dimensions"]
                        """
                        for dimension in confParam['dimensions']:
                            if 'dimensionKey' in dimension:

                                dimensionValue = dimension['dimensionValue']
                                dimensionKey = dimension['dimensionKey']
                                print(f"Selector: {selectValue}, Dimension
Key: {dimensionKey}, Dimension Value: {dimensionValue}")
                                dimensionKeyValueMap[dimensionKey] =
dimensionValue
                        """
                    return dimensionKeyValueMap

except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Agreement with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

```

```
        helper.pretty_print_datetime(get_agreement_information(mp_client,
AGREEMENT_ID))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the payment schedule of an agreement using an AWS SDK

The following code examples show how to get the payment schedule of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
```

```
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;
import
    software.amazon.awssdk.services.marketplaceagreement.model.PaymentScheduleTerm;
import software.amazon.awssdk.services.marketplaceagreement.model.ScheduleItem;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsPaymentSchedule {

    /**
     * Obtain the payment schedule I have agreed to with the agreement, including
     * the invoice date and invoice amount
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        List<Map<String, Object>> paymentScheduleArray =
            getPaymentSchedules(agreementId);

        ReferenceCodesUtils.formatOutput(paymentScheduleArray);
    }

    public static List<Map<String, Object>> getPaymentSchedules(String agreementId)
    {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
            GetAgreementTermsRequest.builder().agreementId(agreementId)
                .build();

        GetAgreementTermsResponse getAgreementTermsResponse =
            marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);
    }
}
```

```
List<Map<String, Object>> paymentScheduleArray = new ArrayList<>();

String currencyCode = "";

for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
    if (acceptedTerm.paymentScheduleTerm() != null) {
        PaymentScheduleTerm paymentScheduleTerm = acceptedTerm.paymentScheduleTerm();
        if (paymentScheduleTerm.currencyCode() != null) {
            currencyCode = paymentScheduleTerm.currencyCode();
        }
        if (paymentScheduleTerm.hasSchedule()) {
            for (ScheduleItem schedule : paymentScheduleTerm.schedule()) {
                if (schedule.chargeDate() != null) {
                    String chargeDate = schedule.chargeDate().toString();
                    String chargeAmount = schedule.chargeAmount();
                    Map<String, Object> scheduleMap = new HashMap<>();
                    scheduleMap.put(ATTRIBUTE_CURRENCY_CODE, currencyCode);
                    scheduleMap.put(ATTRIBUTE_CHARGE_DATE, chargeDate);
                    scheduleMap.put(ATTRIBUTE_CHARGE_AMOUNT, chargeAmount);
                    paymentScheduleArray.add(scheduleMap);
                }
            }
        }
    }
}
return paymentScheduleArray;
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.


```
        sample_file = os.path.join(os.path.dirname(__file__),
SAMPLE_FILE_NAME)
        terms = open_json_file(sample_file)
    else:
        terms = mp_client.get_agreement_terms(agreementId=entity_id)

    paymentScheduleArray = []
    currencyCode = ""
    for term in terms[ROOT_ELEM]:
        if TERM_NAME in term:
            paymentSchedule = term[TERM_NAME]
            if "currencyCode" in paymentSchedule:
                currencyCode = paymentSchedule["currencyCode"]
            if "schedule" in paymentSchedule:
                for sch in paymentSchedule["schedule"]:
                    if "chargeDate" in sch:
                        chargeDate = sch["chargeDate"]
                        chargeAmount = sch["chargeAmount"]
                        # print(f"chargeDate: {chargeDate}, chargeAmount:
{chargeAmount}")

                        schedule = {
                            "currencyCode": currencyCode,
                            "chargeDate": chargeDate,
                            "chargeAmount": chargeAmount,
                        }
                        paymentScheduleArray.append(schedule)

    return paymentScheduleArray

except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Agreement with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")
```

```
    helper.pretty_print_datetime(get_agreement_information(mp_client,
    AGREEMENT_ID))

    # open json file from path

def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the pricing per dimension in an agreement using an AWS SDK

The following code examples show how to get the pricing per dimension in an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
```



```
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;

import java.util.ArrayList;
import java.util.List;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsPricingEachDimension {

    /*
     * Obtain pricing per each dimension in the agreement
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        List<Object> dimensions = getDimensions(agreementId);

        ReferenceCodesUtils.formatOutput(dimensions);
    }

    public static List<Object> getDimensions(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
            GetAgreementTermsRequest.builder().agreementId(agreementId)
                .build();

        GetAgreementTermsResponse getAgreementTermsResponse =
            marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);
    }
}
```

```
List<Object> dimensions = new ArrayList<Object>();

for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
    List<Object> rateInfo = new ArrayList<Object>();
    if (acceptedTerm.configurableUpfrontPricingTerm() != null) {
        if (acceptedTerm.configurableUpfrontPricingTerm().type() != null) {
            rateInfo.add(acceptedTerm.configurableUpfrontPricingTerm().type());
        }
        if (acceptedTerm.configurableUpfrontPricingTerm().currencyCode() != null) {
            rateInfo.add(acceptedTerm.configurableUpfrontPricingTerm().currencyCode());
        }
        if (acceptedTerm.configurableUpfrontPricingTerm().hasRateCards()) {
            rateInfo.add(acceptedTerm.configurableUpfrontPricingTerm().rateCards());
        }
        dimensions.add(rateInfo);
    }
}
return dimensions;
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain pricing per each dimension in the agreement
AG-29
"""
```

```
import json
import logging
import os

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

# agreement id
AGREEMENT_ID = "agmt-11111111111111111111111111111111"

# to use sample file or not
USE_SAMPLE_FILE = False
SAMPLE_FILE_NAME = "mockup_agreement_terms.json"

# attribute name
ROOT_ELEM = "acceptedTerms"
TERM_NAME = "configurableUpfrontPricingTerm"
CONFIG_ELEM = "configuration"
ATTRIBUTE_NAME = "selectorValue"

TERMS_TO_SEARCH = [
    "configurableUpfrontPricingTerm",
    "usageBasedPricingTerm",
    "fixedUpfrontPricingTerm",
]

def get_agreement_information(mp_client, entity_id):
    """
    Returns customer AWS Account id about a given agreement
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of agreement information
    """

    try:
        if USE_SAMPLE_FILE:
            sample_file = os.path.join(os.path.dirname(__file__),
SAMPLE_FILE_NAME)
            terms = open_json_file(sample_file)
        else:
            terms = mp_client.get_agreement_terms(agreementId=entity_id)
```

```
    dimentions = []
    for term in terms[ROOT_ELEM]:
        for t in TERMS_TO_SEARCH:
            rateInfo = []
            if t in term:
                if "type" in term[t]:
                    rateInfo.append(term[t]["type"])
                if "currencyCode" in term[t]:
                    rateInfo.append(term[t]["currencyCode"])
                if "rateCards" in term[t]:
                    rateInfo.append(term[t]["rateCards"])
            dimentions.append(rateInfo)
    return dimentions

except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Agreement with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

    helper.pretty_print_datetime(get_agreement_information(mp_client,
AGREEMENT_ID))

    # open json file from path

def open_json_file(filename):
    with open(filename, "r") as f:
        return json.load(f)

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the pricing type of an agreement using an AWS SDK

The following code examples show how to get the pricing type of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
```

```
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import com.fasterxml.jackson.annotation.JsonAutoDetect.Visibility;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.Objects;
import java.util.Set;

import org.apache.commons.lang3.tuple.Triple;

import
    software.amazon.awssdk.services.marketplacecatalog.MarketplaceCatalogClient;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityRequest;
import
    software.amazon.awssdk.services.marketplacecatalog.model.DescribeEntityResponse;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.fasterxml.jackson.annotation.PropertyAccessor;
import com.fasterxml.jackson.core.JsonProcessingException;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;
import com.fasterxml.jackson.databind.ObjectWriter;
import com.fasterxml.jackson.datatype.jsr310.JavaTimeModule;

/*
 * Obtain the pricing type of the agreement (contract, FPS, metered, free etc.)
 */
public class GetAgreementPricingType {

    private static final String FILTER_NAME = "OfferId";

    private static final String FILTER_VALUE = OFFER_ID;

    // Product types
    private static final String SAAS_PRODUCT = "SaaSProduct";
    private static final String AMI_PRODUCT = "AmiProduct";
```

```
private static final String ML_PRODUCT = "MachineLearningProduct";
private static final String CONTAINER_PRODUCT = "ContainerProduct";
private static final String DATA_PRODUCT = "DataProduct";
private static final String PROSERVICE_PRODUCT = "ProfessionalServicesProduct";
private static final String AIQ_PRODUCT = "AiqProduct";

// Pricing types
private static final String CCP = "CCP";
private static final String ANNUAL = "Annual";
private static final String CONTRACT = "Contract";
private static final String SFT = "SaaS Free Trial";
private static final String HMA = "Hourly and Monthly Agreements";
private static final String HOURLY = "Hourly";
private static final String MONTHLY = "Monthly";
private static final String AFPS = "Annual FPS";
private static final String CFPS = "Contract FPS";
private static final String CCPFPS = "CCP with FPS";
private static final String BYOL = "BYOL";
private static final String FREE = "Free";
private static final String FTH = "Free Trials and Hourly";

// Agreement term pricing types
private static final Set<String> LEGAL = Set.of("LegalTerm");
private static final Set<String> CONFIGURABLE_UPFRONT =
Set.of("ConfigurableUpfrontPricingTerm");
private static final Set<String> USAGE_BASED = Set.of("UsageBasedPricingTerm");
private static final Set<String> CONFIGURABLE_UPFRONT_AND_USAGE_BASED =
Set.of("ConfigurableUpfrontPricingTerm", "UsageBasedPricingTerm");
private static final Set<String> FREE_TRIAL = Set.of("FreeTrialPricingTerm");
private static final Set<String> RECURRING_PAYMENT =
Set.of("RecurringPaymentTerm");
private static final Set<String> USAGE_BASED_AND_RECURRING_PAYMENT =
Set.of("UsageBasedPricingTerm", "RecurringPaymentTerm");
private static final Set<String> FIXED_UPFRONT_AND_PAYMENT_SCHEDULE =
Set.of("FixedUpfrontPricingTerm", "PaymentScheduleTerm");
private static final Set<String>
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED =
Set.of("FixedUpfrontPricingTerm", "PaymentScheduleTerm",
"UsageBasedPricingTerm");
private static final Set<String> BYOL_PRICING = Set.of("ByolPricingTerm");
private static final Set<String> FREE_TRIAL_AND_USAGE_BASED =
Set.of("FreeTrialPricingTerm", "UsageBasedPricingTerm");
```

```
private static final List<Set<String>> ALL_AGREEMENT_TERM_TYPES_COMBINATION
= Arrays.asList(LEGAL, CONFIGURABLE_UPFRONT, USAGE_BASED,
CONFIGURABLE_UPFRONT_AND_USAGE_BASED,
    FREE_TRIAL, RECURRING_PAYMENT, USAGE_BASED_AND_RECURRING_PAYMENT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED, BYOL_PRICING,
FREE_TRIAL_AND_USAGE_BASED);

private static MarketplaceAgreementClient marketplaceAgreementClient =
    MarketplaceAgreementClient.builder()
        .httpClient(ApacheHttpClient.builder().build())
        .credentialsProvider(ProfileCredentialsProvider.create())
        .build();

private static MarketplaceCatalogClient marketplaceCatalogClient =
    MarketplaceCatalogClient.builder()
        .httpClient(ApacheHttpClient.builder().build())
        .credentialsProvider(ProfileCredentialsProvider.create())
        .build();

/*
 * Get agreement Pricing Type given product type, agreement term types and
offer types if needed
 */
public static String getPricingType(String productType, Set<String>
agreementTermType, Set<String> offerType) {
    Map<Triple<String, Set<String>, Set<String>>, String> pricingTypes = new
HashMap<>();

    pricingTypes.put(Triple.of(SAAS_PRODUCT, CONFIGURABLE_UPFRONT_AND_USAGE_BASED,
new HashSet<>()), CCP);
    pricingTypes.put(Triple.of(DATA_PRODUCT, CONFIGURABLE_UPFRONT_AND_USAGE_BASED,
new HashSet<>()), CCP);
    pricingTypes.put(Triple.of(CONTAINER_PRODUCT, CONFIGURABLE_UPFRONT,
CONFIGURABLE_UPFRONT_AND_USAGE_BASED), ANNUAL);
    pricingTypes.put(Triple.of(AMI_PRODUCT, CONFIGURABLE_UPFRONT,
CONFIGURABLE_UPFRONT_AND_USAGE_BASED), ANNUAL);
    pricingTypes.put(Triple.of(ML_PRODUCT, CONFIGURABLE_UPFRONT,
CONFIGURABLE_UPFRONT_AND_USAGE_BASED), ANNUAL);
    pricingTypes.put(Triple.of(CONTAINER_PRODUCT, CONFIGURABLE_UPFRONT,
CONFIGURABLE_UPFRONT), CONTRACT);
    pricingTypes.put(Triple.of(AMI_PRODUCT, CONFIGURABLE_UPFRONT,
CONFIGURABLE_UPFRONT), CONTRACT);
```



```
pricingTypes.put(Triple.of(SAAS_PRODUCT, CONFIGURABLE_UPFRONT, new
HashSet<>()), CONTRACT);
pricingTypes.put(Triple.of(DATA_PRODUCT, CONFIGURABLE_UPFRONT, new
HashSet<>()), CONTRACT);
pricingTypes.put(Triple.of(AIQ_PRODUCT, CONFIGURABLE_UPFRONT, new HashSet<>()),
CONTRACT);
pricingTypes.put(Triple.of(PROSERVICE_PRODUCT, CONFIGURABLE_UPFRONT, new
HashSet<>()), CONTRACT);
pricingTypes.put(Triple.of(SAAS_PRODUCT, FREE_TRIAL, new HashSet<>()), SFT);
pricingTypes.put(Triple.of(AMI_PRODUCT, USAGE_BASED_AND_RECURRING_PAYMENT, new
HashSet<>()), HMA);
pricingTypes.put(Triple.of(SAAS_PRODUCT, USAGE_BASED, new HashSet<>()),
HOURLY);
pricingTypes.put(Triple.of(AMI_PRODUCT, USAGE_BASED, new HashSet<>()), HOURLY);
pricingTypes.put(Triple.of(ML_PRODUCT, USAGE_BASED, new HashSet<>()), HOURLY);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT, RECURRING_PAYMENT, new
HashSet<>()), MONTHLY);
pricingTypes.put(Triple.of(AMI_PRODUCT, RECURRING_PAYMENT, new HashSet<>()),
MONTHLY);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED), AFPS);
pricingTypes.put(Triple.of(AMI_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED), AFPS);
pricingTypes.put(Triple.of(ML_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE, new
HashSet<>()), AFPS);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE, new HashSet<>()), CFPS);
pricingTypes.put(Triple.of(AMI_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE), CFPS);
pricingTypes.put(Triple.of(SAAS_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
new HashSet<>()), CFPS);
pricingTypes.put(Triple.of(DATA_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE,
new HashSet<>()), CFPS);
pricingTypes.put(Triple.of(AIQ_PRODUCT, FIXED_UPFRONT_AND_PAYMENT_SCHEDULE, new
HashSet<>()), CFPS);
pricingTypes.put(Triple.of(PROSERVICE_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE, new HashSet<>()), CFPS);
pricingTypes.put(Triple.of(SAAS_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED, new HashSet<>()), CCPFPS);
pricingTypes.put(Triple.of(DATA_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED, new HashSet<>()), CCPFPS);
pricingTypes.put(Triple.of(AIQ_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED, new HashSet<>()), CCPFPS);
```

```

pricingTypes.put(Triple.of(PROSERVICE_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE_AND_USAGE_BASED, new HashSet<>()), CCPFPS);
pricingTypes.put(Triple.of(AMI_PRODUCT, BYOL_PRICING, new HashSet<>()), BYOL);
pricingTypes.put(Triple.of(SAAS_PRODUCT, BYOL_PRICING, new HashSet<>()), BYOL);
pricingTypes.put(Triple.of(PROSERVICE_PRODUCT, BYOL_PRICING, new HashSet<>()),
BYOL);
pricingTypes.put(Triple.of(AIQ_PRODUCT, BYOL_PRICING, new HashSet<>()), BYOL);
pricingTypes.put(Triple.of(ML_PRODUCT, BYOL_PRICING, new HashSet<>()), BYOL);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT, BYOL_PRICING, new HashSet<>()),
BYOL);
pricingTypes.put(Triple.of(DATA_PRODUCT, BYOL_PRICING, new HashSet<>()), BYOL);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT, LEGAL, new HashSet<>()), FREE);
pricingTypes.put(Triple.of(AMI_PRODUCT, FREE_TRIAL_AND_USAGE_BASED, new
HashSet<>()), FTH);
pricingTypes.put(Triple.of(CONTAINER_PRODUCT, FREE_TRIAL_AND_USAGE_BASED, new
HashSet<>()), FTH);
pricingTypes.put(Triple.of(ML_PRODUCT, FREE_TRIAL_AND_USAGE_BASED, new
HashSet<>()), FTH);

Triple<String, Set<String>, Set<String>> key = Triple.of(productType,
agreementTermType, offerType);

if (pricingTypes.containsKey(key)) {
    return pricingTypes.get(key);
} else {
    return "Unknown";
}
}

/*
 * Given product type and agreement term types, some combinations need to check
offer term types as well.
 */
public static String needToCheckOfferTermsType(String productType, Set<String>
agreementTermTypes) {
    Map<KeyPair, String> offerTermTypes = new HashMap<>();
    offerTermTypes.put(new KeyPair(CONTAINER_PRODUCT, CONFIGURABLE_UPFRONT), "Y");
    offerTermTypes.put(new KeyPair(AMI_PRODUCT, CONFIGURABLE_UPFRONT), "Y");
    offerTermTypes.put(new KeyPair(CONTAINER_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE), "Y");
    offerTermTypes.put(new KeyPair(AMI_PRODUCT,
FIXED_UPFRONT_AND_PAYMENT_SCHEDULE), "Y");

    KeyPair key = new KeyPair(productType, agreementTermTypes);

```

```
    if (offerTermTypes.containsKey(key)) {
        return offerTermTypes.get(key);
    } else {
        return null;
    }
}

public static List<AgreementViewSummary> getAgreementsById() {

    List<AgreementViewSummary> agreementSummaryList = new
    ArrayList<AgreementViewSummary>();

    Filter partyType =
    Filter.builder().name(PARTY_TYPE_FILTER_NAME).values(PARTY_TYPE_FILTER_VALUE_PROPOSER).b

    Filter agreementType =
    Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME).values(AGREEMENT_TYPE_FILTER_VALUE_PUR

    Filter customizeFilter =
    Filter.builder().name(FILTER_NAME).values(FILTER_VALUE).build();

    SearchAgreementsRequest searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(partyType, agreementType, customizeFilter).build();

    SearchAgreementsResponse searchResultResponse =
    marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

    agreementSummaryList.addAll(searchResultResponse.agreementViewSummaries());

    while (searchResultResponse.nextToken() != null &&
    searchResultResponse.nextToken().length() > 0) {
        searchAgreementsRequest =
        SearchAgreementsRequest.builder().catalog(AWS_MP_CATALOG)
            .filters(partyType,
            agreementType).nextToken(searchResultResponse.nextToken()).build();
        searchResultResponse =
        marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);
        agreementSummaryList.addAll(searchResultResponse.agreementViewSummaries());
    }
    return agreementSummaryList;
}
```

```
static class KeyPair {
    private final String first;
    private final Set<String> second;

    public KeyPair(String productType, Set<String> second) {
        this.first = productType;
        this.second = second;
    }

    @Override
    public int hashCode() {
        return Objects.hash(first, second);
    }

    @Override
    public boolean equals(Object obj) {
        if (this == obj)
            return true;
        if (obj == null || getClass() != obj.getClass())
            return false;
        KeyPair other = (KeyPair) obj;
        return Objects.equals(first, other.first) && Objects.equals(second,
other.second);
    }
}

/**
 * Get all the term types for the offer
 */
public static Set<String> getOfferTermTypes(String offerId) {

    Set<String> offerTermTypes = new HashSet<String>();

    DescribeEntityRequest request =
        DescribeEntityRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .entityId(offerId)
            .build();

    DescribeEntityResponse result =
marketplaceCatalogClient.describeEntity(request);

    String details = result.details();
```

```
try {
    ObjectMapper objectMapper = new ObjectMapper();
    JsonNode rootNode = objectMapper.readTree(details);
    JsonNode termsNode = rootNode.get(ATTRIBUTE_TERMS);

    for (JsonNode termNode : termsNode) {
        if (termNode.get(ATTRIBUTE_TYPE_ENTITY) != null ) {
            offerTermTypes.add(termNode.get(ATTRIBUTE_TYPE_ENTITY).asText());
        }
    }
} catch (Exception e) {
    e.printStackTrace();
}

return offerTermTypes;
}

/*
 * Get all the agreement term types
 */
public static Set<String> getAgreementTermTypes(GetAgreementTermsResponse
agreementTerm) {
    Set<String> agreementTermTypes = new HashSet<String>();
    try {
        for (AcceptedTerm term : agreementTerm.acceptedTerms()) {
            ObjectMapper objectMapper = new ObjectMapper();
            JsonNode termNode = objectMapper.readTree(getJson(term));
            Iterator<Map.Entry<String, JsonNode>> fieldsIterator = termNode.fields();
            while (fieldsIterator.hasNext()) {
                Map.Entry<String, JsonNode> entry = fieldsIterator.next();
                JsonNode value = entry.getValue();
                if (value.isObject() && value.has(ATTRIBUTE_TYPE_AGREEMENT)) {
                    agreementTermTypes.add(value.get(ATTRIBUTE_TYPE_AGREEMENT).asText());
                }
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
    return agreementTermTypes;
}
}
```

```
/*
 * make sure all elements in array2 exist in array1
 */
public static boolean allElementsExist(Set<String> array1, Set<String> array2) {
    for (String element : array2) {
        boolean found = false;
        for (String str : array1) {
            if (element.equals(str)) {
                found = true;
                break;
            }
        }
        if (!found) {
            return false;
        }
    }
    return true;
}

/*
 * Find the combinations of the agreement term types for the agreement
 */
public static Set<String> getMatchedTermTypesCombination(Set<String>
agreementTermTypes) {
    Set<String> matchedCombination = new HashSet<String>();
    for (Set<String> element : ALL_AGREEMENT_TERM_TYPES_COMBINATION) {
        if (allElementsExist(agreementTermTypes, element)) {
            matchedCombination = element;
        }
    }
    return matchedCombination;
}

public static void main(String[] args) {

    List<AgreementViewSummary> agreements = getAgreementsById();

    for (AgreementViewSummary summary : agreements) {
        String pricingType = "";
        String agreementId = summary.agreementId();
        System.out.println(agreementId);
        String offerId = summary.proposalSummary().offerId();
    }
}
```

```
//get all pricing term types for the offer in the agreement
Set<String> offerTermTypes = getOfferTermTypes(offerId);
String productType = summary.proposalSummary().resources().get(0).type();

//get all pricing term types for the agreement
GetAgreementTermsRequest getAgreementTermsRequest =
    GetAgreementTermsRequest.builder().agreementId(agreementId)
        .build();
GetAgreementTermsResponse getAgreementTermsResponse =
marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);
Set<String> agreementTermTypes =
getAgreementTermTypes(getAgreementTermsResponse);

//get matched pricing term type combination set
Set<String> agreementMatchedTermType =
getMatchedTermTypesCombination(agreementTermTypes);

//check to see if this agreement pricing term combination needs additional
check on offer pricing terms
String needToCheckOfferType = needToCheckOfferTermsType(productType,
agreementMatchedTermType);

// get the pricing type for the agreement based on the product type, agreement
term types and offer term types if needed
if (needToCheckOfferType != null) {
    Set<String> offerMatchedTermType =
getMatchedTermTypesCombination(offerTermTypes);
    pricingType = getPricingType(productType, agreementMatchedTermType,
offerMatchedTermType);
} else if (agreementMatchedTermType == LEGAL) {
    pricingType = FREE;
} else {
    pricingType = getPricingType(productType, agreementMatchedTermType, new
HashSet());
}
System.out.println("Pricing type is " + pricingType);
}
}

private static String getJson(Object result) {
    String json = "";

    try {
        ObjectMapper om = new ObjectMapper();
```

```
om.setVisibility(PropertyAccessor.FIELD, Visibility.ANY);
om.registerModule(new JavaTimeModule());
ObjectWriter ow = om.writer().withDefaultPrettyPrinter();

    json = ow.writeValueAsString(result);
} catch (JsonProcessingException e) {
    e.printStackTrace();
}
return json;
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the pricing type of the agreement (contract, FPS, metered, free etc.)
AG-16
"""

import json
import logging

import boto3
from botocore.exceptions import ClientError

# To search by offer id: OfferId; by product id: ResourceIdentifier; by product
type: ResourceType
```



```
idType = "OfferId"

# replace id value as needed
idValue = "offer-11111111111111"

MAX_PAGE_RESULTS = 10

# catalog; switch to AWSMarketplace for release
AWSMPCATALOG = "AWSMarketplace"

# product types

SaaSProduct = "SaaSProduct"
AmiProduct = "AmiProduct"
MLProduct = "MachineLearningProduct"
ContainerProduct = "ContainerProduct"
DataProduct = "DataProduct"
ProServiceProduct = "ProfessionalServicesProduct"
AiqProduct = "AiqProduct"

# Define pricing types
CCP = "CCP"
Annual = "Annual"
Contract = "Contract"
SFT = "SaaS Free Trial"
HMA = "Hourly and Monthly Agreements"
Hourly = "Hourly"
Monthly = "Monthly"
AFPS = "Annual FPS"
CFPS = "Contract FPS"
CCPFPS = "CCP with FPS"
BYOL = "BYOL"
Free = "Free"
FTH = "Free Trials and Hourly"

# Define Agreement Term Types
legal = ["LegalTerm"]
config = ["ConfigurableUpfrontPricingTerm"]
usage = ["UsageBasedPricingTerm"]
config_usage = ["ConfigurableUpfrontPricingTerm", "UsageBasedPricingTerm"]
freeTrial = ["FreeTrialPricingTerm"]
recur = ["RecurringPaymentTerm"]
usage_recur = ("UsageBasedPricingTerm", "RecurringPaymentTerm")
fixed_payment = ["FixedUpfrontPricingTerm", "PaymentScheduleTerm"]
```

```

fixed_payment_usage = [
    "FixedUpfrontPricingTerm",
    "PaymentScheduleTerm",
    "UsageBasedPricingTerm",
]
byol = ["ByolPricingTerm"]
freeTrial_usage = ("FreeTrialPricingTerm", "UsageBasedPricingTerm")
all_agreement_types_combination = (
    legal,
    config,
    usage,
    config_usage,
    freeTrial,
    recur,
    usage_recur,
    fixed_payment,
    fixed_payment_usage,
    byol,
    freeTrial_usage,
)

# get pricing type method given product type, agreement term type and offer type
if needed
def get_pricing_type(product_type, agreement_term_type, offer_type):
    pricing_types = {
        (SaaSProduct, frozenset(config_usage), frozenset("")): CCP,
        (DataProduct, frozenset(config_usage), frozenset("")): CCP,
        (ContainerProduct, frozenset(config), frozenset(config_usage)): Annual,
        (AmiProduct, frozenset(config), frozenset(config_usage)): Annual,
        (MLProduct, frozenset(config), frozenset(config_usage)): Annual,
        (ContainerProduct, frozenset(config), frozenset(config)): Contract,
        (AmiProduct, frozenset(config), frozenset(config)): Contract,
        (SaaSProduct, frozenset(config), frozenset("")): Contract,
        (DataProduct, frozenset(config), frozenset("")): Contract,
        (AiqProduct, frozenset(config), frozenset("")): Contract,
        (ProServiceProduct, frozenset(config), frozenset("")): Contract,
        (SaaSProduct, frozenset(freeTrial), frozenset("")): SFT,
        (AmiProduct, frozenset(usage_recur), frozenset("")): HMA,
        (SaaSProduct, frozenset(usage), frozenset("")): Hourly,
        (AmiProduct, frozenset(usage), frozenset("")): Hourly,
        (ContainerProduct, frozenset(usage), frozenset("")): Hourly,
        (MLProduct, frozenset(usage), frozenset("")): Hourly,
        (ContainerProduct, frozenset(recur), frozenset("")): Monthly,

```

```

        (AmiProduct, frozenset(recur), frozenset("")): Monthly,
        (
            ContainerProduct,
            frozenset(fixed_payment),
            frozenset(fixed_payment_usage),
        ): AFPS,
    (AmiProduct, frozenset(fixed_payment), frozenset(fixed_payment_usage)):
AFPS,
    (MLProduct, frozenset(fixed_payment), frozenset("")): AFPS,
    (ContainerProduct, frozenset(fixed_payment), frozenset(fixed_payment)):
CFPS,
    (AmiProduct, frozenset(fixed_payment), frozenset(fixed_payment)): CFPS,
    (SaaSProduct, frozenset(fixed_payment), frozenset("")): CFPS,
    (DataProduct, frozenset(fixed_payment), frozenset("")): CFPS,
    (AiqProduct, frozenset(fixed_payment), frozenset("")): CFPS,
    (ProServiceProduct, frozenset(fixed_payment), frozenset("")): CFPS,
    (SaaSProduct, frozenset(fixed_payment_usage), frozenset("")): CCPFPS,
    (DataProduct, frozenset(fixed_payment_usage), frozenset("")): CCPFPS,
    (AiqProduct, frozenset(fixed_payment_usage), frozenset("")): CCPFPS,
    (ProServiceProduct, frozenset(fixed_payment_usage), frozenset("")):
CCPFPS,
    (AmiProduct, frozenset(byol), frozenset("")): BYOL,
    (SaaSProduct, frozenset(byol), frozenset("")): BYOL,
    (ProServiceProduct, frozenset(byol), frozenset("")): BYOL,
    (AiqProduct, frozenset(byol), frozenset("")): BYOL,
    (MLProduct, frozenset(byol), frozenset("")): BYOL,
    (ContainerProduct, frozenset(byol), frozenset("")): BYOL,
    (DataProduct, frozenset(byol), frozenset("")): BYOL,
    (ContainerProduct, frozenset(legal), frozenset("")): Free,
    (AmiProduct, frozenset(freeTrial_usage), frozenset("")): FTH,
    (ContainerProduct, frozenset(freeTrial_usage), frozenset("")): FTH,
    (MLProduct, frozenset(freeTrial_usage), frozenset("")): FTH,
}

key = (product_type, agreement_term_type, offer_type)
if key in pricing_types:
    return pricing_types[key]
else:
    return "Unknown"

# Example usage for testing purpose
"""
product_type = SaaSProduct

```

```
agreement_term_type = frozenset(config_usage)
offer_type = frozenset('')
pricing_type = get_pricing_type(product_type, agreement_term_type, offer_type)
print("pricing type = " + pricing_type) # Output: CCP
"""

# check if offer term types are needed; if Y, needed
def get_offer_term_type(product_type, agreement_term_type):
    offer_term_types = {
        (ContainerProduct, frozenset(config)): "Y",
        (AmiProduct, frozenset(config)): "Y",
        (ContainerProduct, frozenset(fixed_payment)): "Y",
        (AmiProduct, frozenset(fixed_payment)): "Y",
        (AmiProduct, frozenset(fixed_payment), frozenset(fixed_payment)): "Y",
    }

    key = (product_type, agreement_term_type)
    if key in offer_term_types:
        return offer_term_types[key]
    else:
        return

logger = logging.getLogger(__name__)

def get_agreements(mp_client):
    AgreementSummaryList = []
    partyTypes = ["Proposer"]
    for value in partyTypes:
        try:
            agreement = mp_client.search_agreements(
                catalog=AWSMPCATALOG,
                maxResults=MAX_PAGE_RESULTS,
                filters=[
                    {"name": "PartyType", "values": [value]},
                    {"name": "idType", "values": [idValue]},
                    {"name": "AgreementType", "values": ["PurchaseAgreement"]},
                ],
            )
        except ClientError as e:
            logger.error("Could not complete search_agreements request.")
            raise
```

```
AgreementSummaryList.extend(agreement["agreementViewSummaries"])

while "nextToken" in agreement and agreement["nextToken"] is not None:
    try:
        agreement = mp_client.search_agreements(
            catalog=AWSMPCATALOG,
            maxResults=MAX_PAGE_RESULTS,
            nextToken=agreement["nextToken"],
            filters=[
                {"name": "PartyType", "values": [value]},
                {"name": idType, "values": [idValue]},
                {"name": "AgreementType", "values":
["PurchaseAgreement"]}],
            ],
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

return AgreementSummaryList

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

    # find all agreements matching the specified idType and idValue
    agreements = get_agreements(mp_client)

    for item in agreements:
        pricingType = ""
        agreement_id = item["agreementId"]

        # get term types inside offer
        offer_term_types = get_offer_term_types(item)
```

```
# even though multiple product types are allowed for one agreement, only
need the first one
productType = item["resourceSummaries"][0]["resourceType"]

# get agreement terms types
agreementTerm = mp_client.get_agreement_terms(agreementId=agreement_id)

agreementTermTypes = get_agreement_term_types(agreementTerm)

# match with agreement term type group
matchedTermType = getMatchedTermTypesCombination(agreementTermTypes)

# check if offer term type is needed.
offer_term_type_needed = get_offer_term_type(
    productType, frozenset(matchedTermType)
)

# get pricing type given product type, agreement term types and offer
type if needed;
# one exception is Container with Legal term. LegalTerm needs to be the
only term present
if offer_term_type_needed is not None:
    matchedOfferTermTypes =
getMatchedTermTypesCombination(offer_term_types)
    print(f"matchedOfferTermType = {matchedOfferTermTypes}")
    pricingType = get_pricing_type(
        productType,
        frozenset(matchedTermType),
        frozenset(matchedOfferTermTypes),
    )
elif set(matchedTermType) == set(legal):
    pricingType = Free
else:
    pricingType = get_pricing_type(
        productType, frozenset(matchedTermType), frozenset("")
    )

print(
    f"agreementId={agreement_id};productType={productType};
agreementTermTypes={agreementTermTypes}; matchedTermType={matchedTermType};
offerTermTypeNeeded={offer_term_type_needed};
offer_term_types={offer_term_types}"
)
print(f"pricing type={pricingType}")
```

```
def getMatchedTermTypesCombination(agreementTermTypes):
    matchedCombination = ()
    for element in all_agreement_types_combination:
        if check_elements(agreementTermTypes, element):
            matchedCombination = element
    return matchedCombination

def get_offer_term_types(item):
    offer_id = item["agreementTokenSummary"]["offerId"]
    mp_catalogAPI_client = boto3.client("marketplace-catalog")
    offer_document = get_entity_information(mp_catalogAPI_client, offer_id)
    offerDetail = offer_document["Details"]
    offerDetail_json_object = json.loads(offerDetail)
    offer_term_types = [term["Type"] for term in
offerDetail_json_object["Terms"]]
    return offer_term_types

# make sure all elements in array2 exist in array1
def check_elements(array1, array2):
    for element in array2:
        if element not in array1:
            return False
    return True

def get_entity_information(mp_client, entity_id):
    """
    Returns information about a given entity
    Args: entity_id str: Entity to return
    Returns: dict: Dictionary of entity information
    """

    try:
        response = mp_client.describe_entity(
            Catalog="AWSMarketplace",
            EntityId=entity_id,
        )

        return response
```

```
except ClientError as e:
    if e.response["Error"]["Code"] == "ResourceNotFoundException":
        logger.error("Entity with ID %s not found.", entity_id)
    else:
        logger.error("Unexpected error: %s", e)

def get_agreement_term_types(agreementTerm):
    types = []
    for term in agreementTerm["acceptedTerms"]:
        for value in term.values():
            if isinstance(value, dict) and "type" in value:
                types.append(value["type"])
    return types

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the product type of an agreement using an AWS SDK

The following code examples show how to get the product type of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
```



```
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;
import software.amazon.awssdk.services.marketplaceagreement.model.Resource;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import java.util.ArrayList;
import java.util.List;

import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementProductType {

    /**
     * Obtain the Product Type of the product the agreement was created on
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        List<String> productIds = getProducts(agreementId);

        ReferenceCodesUtils.formatOutput(productIds);
    }

    public static List<String> getProducts(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        DescribeAgreementRequest describeAgreementRequest =
            DescribeAgreementRequest.builder()
                .agreementId(agreementId)
```

```
.build();

DescribeAgreementResponse describeAgreementResponse =
marketplaceAgreementClient.describeAgreement(describeAgreementRequest);

List<String> productIds = new ArrayList<String>();
for (Resource resource :
describeAgreementResponse.proposalSummary().resources()) {
    productIds.add(resource.id() + ":" + resource.type());
}
return productIds;
}
}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the Product Type of the product the agreement was created on
AG-11
"""

import logging

import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)
```



```
if __name__ == "__main__":  
    usage_demo()
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the status of an agreement using an AWS SDK

The following code examples show how to get the status of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.  
// SPDX-License-Identifier: Apache-2.0  
package com.example.awsmarketplace.agreementapi;  
  
import static  
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;  
  
import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;  
import software.amazon.awssdk.http.apache.ApacheHttpClient;  
import  
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;  
import  
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementRequest;  
import  
    software.amazon.awssdk.services.marketplaceagreement.model.DescribeAgreementResponse;
```

```
public class GetAgreementStatus {

    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        DescribeAgreementResponse describeAgreementResponse =
            getDescribeAgreementResponse(agreementId);

        System.out.println("Agreement status is " +
            describeAgreementResponse.status());

    }

    public static DescribeAgreementResponse getDescribeAgreementResponse(String
        agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        DescribeAgreementRequest describeAgreementRequest =
            DescribeAgreementRequest.builder()
                .agreementId(agreementId)
                .build();

        DescribeAgreementResponse describeAgreementResponse =
            marketplaceAgreementClient.describeAgreement(describeAgreementRequest);
        return describeAgreementResponse;
    }

}
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get all agreement status
AG-13

Example Usage: python3 get_agreement_status.py --agreement-id <agreement-id>
"""

import argparse
import logging

import boto3
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")

logger = logging.getLogger(__name__)

def get_agreement(agreement_id):
    try:
        response = mp_client.describe_agreement(agreementId=agreement_id)
        return response
    except ClientError as e:
        logger.error(f"Could not complete search_agreements request. {e}")

    return None

if __name__ == "__main__":
```

```
parser = argparse.ArgumentParser()
parser.add_argument(
    "--agreement-id",
    "-aid",
    help="Provide agreement ID to describe agreement status",
    required=True,
)
args = parser.parse_args()

response = get_agreement(agreement_id=args.agreement_id)

if response is not None:
    print(f"Agreement status: {response['status']}")
else:
    print(f"No agreement found for {args.agreement_id}")
```

- For API details, see [DescribeAgreement](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the support terms of an agreement using an AWS SDK

The following code examples show how to get the support terms of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;
```

```
import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import software.amazon.awssdk.services.marketplaceagreement.model.AcceptedTerm;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;
import software.amazon.awssdk.services.marketplaceagreement.model.SupportTerm;

import java.util.ArrayList;
import java.util.List;

import static
    com.example.awsmarketplace.utils.ReferenceCodesConstants.AGREEMENT_ID;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class GetAgreementTermsSupportTerm {

    /**
     * Obtain the support and refund policy I have provided to the customer
     */
    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        List<SupportTerm> supportTerms = getSupportTerms(agreementId);

        ReferenceCodesUtils.formatOutput(supportTerms);
    }

    public static List<SupportTerm> getSupportTerms(String agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
            GetAgreementTermsRequest.builder().agreementId(agreementId)
                .build();
```



```
GetAgreementTermsResponse getAgreementTermsResponse =
marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);

List<SupportTerm> supportTerms = new ArrayList<>();

for (AcceptedTerm acceptedTerm : getAgreementTermsResponse.acceptedTerms()) {
    if (acceptedTerm.supportTerm() != null) {
        supportTerms.add(acceptedTerm.supportTerm());
    }
}
return supportTerms;
}
}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Obtain the support and refund policy I have provided to the customer for an
agreement
AG-19

Example Usage: python3 get_agreement_support_terms.py --agreement-id <agreement-
id>
"""

import argparse
import logging
```

```
import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)

mp_client = boto3.client("marketplace-agreement")

def get_agreement_terms(agreement_id):
    try:
        agreement = mp_client.get_agreement_terms(agreementId=agreement_id)
        return agreement

    except ClientError as e:
        if e.response["Error"]["Code"] == "ResourceNotFoundException":
            logger.error("Agreement with ID %s not found.", agreement_id)

        else:
            logger.error("Unexpected error: %s", e)

    return None

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--agreement-id",
        "-aid",
        help="Provide agreement ID to describe agreement status",
        required=True,
    )
    args = parser.parse_args()

    agreement = get_agreement_terms(agreement_id=args.agreement_id)

    if agreement is not None:
        support_found = False

        for term in agreement["acceptedTerms"]:
            if "supportTerm" in term.keys():
                helper.pretty_print_datetime(term)
                support_found = True
```

```
if not support_found:
    print(f"No support term found for agreement: {args.agreement_id}")
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Get the terms of an agreement using an AWS SDK

The following code example shows how to get the terms of an agreement.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.GetAgreementTermsResponse;
```

```
public class GetAgreementTerms {

    public static void main(String[] args) {

        String agreementId = args.length > 0 ? args[0] : AGREEMENT_ID;

        GetAgreementTermsResponse getAgreementTermsResponse =
            getAgreementTermsResponse(agreementId);

        ReferenceCodesUtils.formatOutput(getAgreementTermsResponse);

    }

    public static GetAgreementTermsResponse getAgreementTermsResponse(String
agreementId) {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        GetAgreementTermsRequest getAgreementTermsRequest =
            GetAgreementTermsRequest.builder()
                .agreementId(agreementId)
                .build();

        GetAgreementTermsResponse getAgreementTermsResponse =
            marketplaceAgreementClient.getAgreementTerms(getAgreementTermsRequest);
        return getAgreementTermsResponse;
    }

}
```

- For API details, see [GetAgreementTerms](#) in *AWS SDK for Java 2.x API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by account ID using an AWS SDK

The following code example shows how to search for agreements by account ID.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to get agreement by customer AWS
account ID
AG-02
"""

import argparse
import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")
logger = logging.getLogger(__name__)

MAX_PAGE_RESULTS = 10

def get_agreements(account_id):
    AgreementSummaryList = []

    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
```

```
        maxResults=MAX_PAGE_RESULTS,
        filters=[
            {"name": "PartyType", "values": ["Proposer"]},
            {"name": "AcceptorId", "values": [account_id]},
            {"name": "AgreementType", "values": ["PurchaseAgreement"]},
        ],
    )
except ClientError as e:
    logger.error("Could not complete search_agreements request.")
    raise e

AgreementSummaryList.extend(agreement["agreementViewSummaries"])

while "nextToken" in agreement and agreement["nextToken"] is not None:
    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            nextToken=agreement["nextToken"],
            filters=[
                {"name": "PartyType", "values": ["Proposer"]},
                {"name": "AcceptorId", "values": [account_id]},
                {"name": "AgreementType", "values": ["PurchaseAgreement"]},
            ],
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise e

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

return AgreementSummaryList

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "--account_id",
        "-aid",
        help="Provide accepting account ID to search for agreements",
        required=True,
    )
    args = parser.parse_args()
```

```
response = get_agreements(account_id=args.account_id)

helper.pretty_print_datetime(response)
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by agreement ID using an AWS SDK

The following code example shows how to search for agreements by agreement ID.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to search for agreements give id
information
AG-02-A
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError
```

```
# To search by offer id: OfferId; by product id: ResourceIdentifier; by product
type: ResourceType
idType = "ResourceType"

# replace id value as needed
idValue = "SaaSProduct"

MAX_PAGE_RESULTS = 10

logger = logging.getLogger(__name__)

def get_agreements(mp_client):
    AgreementSummaryList = []
    partyTypes = ["Proposer"]
    for value in partyTypes:
        try:
            agreement = mp_client.search_agreements(
                catalog="AWSMarketplace",
                maxResults=MAX_PAGE_RESULTS,
                filters=[
                    {"name": "PartyType", "values": [value]},
                    {"name": idType, "values": [idValue]},
                    {"name": "AgreementType", "values": ["PurchaseAgreement"]},
                ],
            )
        except ClientError as e:
            logger.error("Could not complete search_agreements request.")
            raise e

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

    while "nextToken" in agreement and agreement["nextToken"] is not None:
        try:
            agreement = mp_client.search_agreements(
                catalog="AWSMarketplace",
                maxResults=MAX_PAGE_RESULTS,
                nextToken=agreement["nextToken"],
                filters=[
                    {"name": "PartyType", "values": [value]},
                    {"name": idType, "values": [idValue]},
                    {"name": "AgreementType", "values":
["PurchaseAgreement"]},
                ],
            )
```



```
        )
        except ClientError as e:
            logger.error("Could not complete search_agreements request.")
            raise e

        AgreementSummaryList.extend(agreement["agreementViewSummaries"])

    return AgreementSummaryList

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement in the AWS Marketplace Catalog.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

    helper.pretty_print_datetime(get_agreements(mp_client))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by end date using an AWS SDK

The following code examples show how to search for agreements by end date.

Java

SDK for Java 2.x

 **Note**

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

public class SearchAgreementsByEndDate {

    static String beforeOrAfterEndTimeFilterName =
        BeforeOrAfterEndTimeFilterName.BeforeEndTime.name();

    static String cutoffDate = "2050-11-18T00:00:00Z";

    static String partyTypeFilterValue = PARTY_TYPE_FILTER_VALUE_PROPOSER;
```

```
public static void main(String[] args) {

    List<AgreementViewSummary> agreementSummaryList = getAgreements();

    ReferenceCodesUtils.formatOutput(agreementSummaryList);
}

public static List<AgreementViewSummary> getAgreements() {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    // set up filters

    Filter partyTypeFilter = Filter.builder().name(PARTY_TYPE_FILTER_NAME)
        .values(PARTY_TYPE_FILTER_VALUE_PROPOSER).build();

    Filter agreementTypeFilter = Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME)
        .values(AGREEMENT_TYPE_FILTER_VALUE_PURCHASEAGREEMENT).build();

    Filter customizeFilter =
        Filter.builder().name(beforeOrAfterEndtimeFilterName).values(cutoffDate).build();

    List<Filter> filters = new ArrayList<Filter>();

    filters.addAll(Arrays.asList(partyTypeFilter, agreementTypeFilter,
        customizeFilter));

    // search agreement with filters

    SearchAgreementsRequest searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(filters)
            .build();

    SearchAgreementsResponse searchAgreementResponse=
        marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

    List<AgreementViewSummary> agreementSummaryList = new
        ArrayList<AgreementViewSummary>();
}
```

```
agreementSummaryList.addAll(searchAgreementResponse.agreementViewSummaries());

while (searchAgreementResponse.nextToken() != null &&
searchAgreementResponse.nextToken().length() > 0) {
    searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(filters)
            .nextToken(searchAgreementResponse.nextToken())
            .build();
    searchAgreementResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);
    agreementSummaryList.addAll(searchAgreementResponse.agreementViewSummaries());
}
return agreementSummaryList;
}
}
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Java 2.x API Reference*.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to search for agreement
information before or after end date
AG-03
"""

import logging
```

```
import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")

# change to 'AfterEndTime' if after endtime is desired
beforeOrAfterEndtimeFilterName = "BeforeEndTime"

# Make sure to use the same date format as below
cutoffDate = "2322-11-18T00:00:00Z"

MAX_PAGE_RESULTS = 10

logger = logging.getLogger(__name__)

def get_agreements():
    AgreementSummaryList = []

    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            filters=[
                {"name": "PartyType", "values": ["Proposer"]},
                {"name": beforeOrAfterEndtimeFilterName, "values": [cutoffDate]},
                {"name": "AgreementType", "values": ["PurchaseAgreement"]},
            ],
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

    while "nextToken" in agreement:
        try:
            agreement = mp_client.search_agreements(
                catalog="AWSMarketplace",
                maxResults=MAX_PAGE_RESULTS,
                nextToken=agreement["nextToken"],
                filters=[
```

```
        {"name": "PartyType", "values": ["Proposer"]},
        {
            "name": beforeOrAfterEndtimeFilterName,
            "values": [cutoffDate],
        },
        {"name": "AgreementType", "values": ["PurchaseAgreement"]},
    ],
)
except ClientError as e:
    logger.error("Could not complete search_agreements request.")
    raise

AgreementSummaryList.extend(agreement["agreementViewSummaries"])

return AgreementSummaryList

if __name__ == "__main__":
    agreements = get_agreements()
    helper.pretty_print_datetime(agreements)
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by offer ID using an AWS SDK

The following code example shows how to search for agreements by offer ID.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to search for agreements by offer
id
AG-0
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

# offer id to search by
offerId = "11111111111111111111111111111111"

MAX_PAGE_RESULTS = 10

logger = logging.getLogger(__name__)

def get_agreements(mp_client):
    AgreementSummaryList = []
    partyTypes = ["Proposer"]
    for value in partyTypes:
        try:
            agreement = mp_client.search_agreements(
                catalog="AWSMarketplace",
                maxResults=MAX_PAGE_RESULTS,
                filters=[
                    {"name": "PartyType", "values": [value]},
                    {"name": "OfferId", "values": [offerId]},
                    {"name": "AgreementType", "values": ["PurchaseAgreement"]},
                ],
            )
        except ClientError as e:
            logger.error("Could not complete search_agreements request.")
            raise

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])
```

```
while "nextToken" in agreement and agreement["nextToken"] is not None:
    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            nextToken=agreement["nextToken"],
            filters=[
                {"name": "PartyType", "values": [value]},
                {"name": "OfferId", "values": [offerId]},
                {"name": "AgreementType", "values":
["PurchaseAgreement"]}],
            ],
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

return AgreementSummaryList

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Looking for an agreement by offer id.")
    print("-" * 88)

    mp_client = boto3.client("marketplace-agreement")

    helper.pretty_print_datetime(get_agreements(mp_client))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by product ID using an AWS SDK

The following code example shows how to search for agreements by product ID.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to search for agreement by
product id
AG-02
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

# product id to search by
resourceId = "prod-11111111111111"

MAX_PAGE_RESULTS = 10

logger = logging.getLogger(__name__)

def get_agreements(mp_client):
```

```
AgreementSummaryList = []
partyTypes = ["Proposer"]
for value in partyTypes:
    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            filters=[
                {"name": "PartyType", "values": [value]},
                {"name": "ResourceIdentifier", "values": [resourceId]},
                {"name": "AgreementType", "values": ["PurchaseAgreement"]},
            ],
        )
    except ClientError as e:
        logger.error("Could not complete list_entities request.")
        raise

AgreementSummaryList.extend(agreement["agreementViewSummaries"])

while "nextToken" in agreement:
    try:
        agreement = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            nextToken=agreement["nextToken"],
            filters=[
                {"name": "PartyType", "values": [value]},
                {"name": "ResourceIdentifier", "values": [resourceId]},
                {"name": "AgreementType", "values":
["PurchaseAgreement"]},
            ],
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise

    AgreementSummaryList.extend(agreement["agreementViewSummaries"])

return AgreementSummaryList

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")
```

```
print("-" * 88)
print("Looking for an agreement in the AWS Marketplace Catalog.")
print("-" * 88)

mp_client = boto3.client("marketplace-agreement")

helper.pretty_print_datetime(get_agreements(mp_client))

if __name__ == "__main__":
    usage_demo()
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements by status using an AWS SDK

The following code example shows how to search for agreements by status.

Python

SDK for Python (Boto3)

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0
"""
Purpose
Shows how to use the AWS SDK for Python (Boto3) to filter agreements by status
AG-04

Example Usage: python3 search_agreements_by_status.py
```

```
"""

import logging

import boto3
import utils.helpers as helper
from botocore.exceptions import ClientError

mp_client = boto3.client("marketplace-agreement")

logger = logging.getLogger(__name__)

MAX_PAGE_RESULTS = 10

party_type_list = ["Proposer"]
agreement_type_list = ["PurchaseAgreement"]

# Accepted values: "ACTIVE", "TERMINATED", "CANCELED", "EXPIRED", "REPLACED",
# "RENEWED"
status_list = ["ACTIVE"]

filter_list = [
    {"name": "PartyType", "values": party_type_list},
    {"name": "AgreementType", "values": agreement_type_list},
    {"name": "Status", "values": status_list},
]

agreement_results_list = []

def get_agreements(filter_list=filter_list):
    try:
        agreements = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            filters=filter_list,
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise e

    agreement_results_list.extend(agreements["agreementViewSummaries"])

    while "nextToken" in agreements and agreements["nextToken"] is not None:
```

```
    try:
        agreements = mp_client.search_agreements(
            catalog="AWSMarketplace",
            maxResults=MAX_PAGE_RESULTS,
            nextToken=agreements["nextToken"],
            filters=filter_list,
        )
    except ClientError as e:
        logger.error("Could not complete search_agreements request.")
        raise e

    agreement_results_list.extend(agreements["agreementViewSummaries"])

    helper.pretty_print_datetime(agreement_results_list)
    return agreement_results_list

if __name__ == "__main__":
    agreements_list = get_agreements(filter_list)
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements with one custom filter using an AWS SDK

The following code example shows how to search for agreements with one custom filter.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import com.example.awsmarketplace.utils.ReferenceCodesUtils;

/**
 * To search by
 * offer id: OfferId;
 * product id: ResourceIdentifier;
 * customer AWS account id: AcceptorAccountId
 * product type: ResourceType (i.e. SaaSProduct)
 * status: Status. status values can be: ACTIVE, CANCELED,
 * EXPIRED, RENEWED, REPLACED, ROLLED_BACK, SUPERSEDED, TERMINATED
 */
public class SearchAgreementsByOneFilter {

    private static final String FILTER_NAME = "ResourceType";

    private static final String FILTER_VALUE = "SaaSProduct";

    /**
     * search agreements by one customize filter
     */
}
```

```
public static void main(String[] args) {

    List<AgreementViewSummary> agreementSummaryList = getAgreements();

    ReferenceCodesUtils.formatOutput(agreementSummaryList);
}

public static List<AgreementViewSummary> getAgreements() {
    MarketplaceAgreementClient marketplaceAgreementClient =
        MarketplaceAgreementClient.builder()
            .httpClient(ApacheHttpClient.builder().build())
            .credentialsProvider(ProfileCredentialsProvider.create())
            .build();

    Filter partyTypeFilter = Filter.builder().name(PARTY_TYPE_FILTER_NAME)
        .values(PARTY_TYPE_FILTER_VALUE_PROPOSER).build();

    Filter agreementTypeFilter = Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME)
        .values(AGREEMENT_TYPE_FILTER_VALUE_PURCHASEAGREEMENT).build();

    Filter customizeFilter =
        Filter.builder().name(FILTER_NAME).values(FILTER_VALUE).build();

    List<Filter> filters = new ArrayList<Filter>();

    filters.addAll(Arrays.asList(partyTypeFilter, agreementTypeFilter,
        customizeFilter));

    SearchAgreementsRequest searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(filters)
            .build();

    SearchAgreementsResponse searchAgreementsResponse =
        marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

    List<AgreementViewSummary> agreementSummaryList = new
        ArrayList<AgreementViewSummary>();

    agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());

    while (searchAgreementsResponse.nextToken() != null &&
        searchAgreementsResponse.nextToken().length() > 0) {
        searchAgreementsRequest =
```

```
SearchAgreementsRequest.builder()
    .catalog(AWS_MP_CATALOG)
    .filters(filters)
    .nextToken(searchAgreementsResponse.nextToken())
    .build();
searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());
}
return agreementSummaryList;
}
}
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Java 2.x API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Search for agreements with two custom filters using an AWS SDK

The following code example shows how to search for agreements with two custom filters.

Java

SDK for Java 2.x

Note

There's more on GitHub. Find the complete example and learn how to set up and run in the [AWS Marketplace API Reference Code Library](#) repository.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
package com.example.awsmarketplace.agreementapi;

import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
```



```
import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;
import software.amazon.awssdk.http.SdkHttpClient;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import
    software.amazon.awssdk.services.marketplaceagreement.MarketplaceAgreementClient;
import
    software.amazon.awssdk.services.marketplaceagreement.model.AgreementViewSummary;
import software.amazon.awssdk.services.marketplaceagreement.model.Filter;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsRequest;
import
    software.amazon.awssdk.services.marketplaceagreement.model.SearchAgreementsResponse;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import static com.example.awsmarketplace.utils.ReferenceCodesConstants.*;
import com.example.awsmarketplace.utils.ReferenceCodesUtils;

/**
 * Party Type = Proposer AND Acceptor:
 *   AfterEndTime
 *   BeforeEndTime
 *   ResourceIdentifier + BeforeEndTime
 *   ResourceIdentifier + AfterEndTime
 *   ResourceType + BeforeEndTime
 *   ResourceType + AfterEndTime
 *
 * Party Type = Proposer
 *   ResourceIdentifier
 *   OfferId
 *   AcceptorAccountId
 *   Status (ACTIVE)
 *   Status (ACTIVE) + ResourceIdentifier
 *   Status (ACTIVE) + AcceptorAccountId
 *   Status (ACTIVE) + OfferId
 *   Status (ACTIVE) + ResourceType
 *   AcceptorAccountId + BeforeEndTime
 *   AcceptorAccountId + AfterEndTime
 *   AcceptorAccountId + AfterEndTime
 *   OfferId + BeforeEndTime
 *

```

```
* Status values can be: ACTIVE, CANCELLED, EXPIRED, RENEWED, REPLACED,
ROLLED_BACK, SUPERSEDED, TERMINATED
*/

public class SearchAgreementsByTwoFilters {

    public static final String FILTER_1_NAME = "ResourceType";

    public static final String FILTER_1_VALUE = "SaaSProduct";

    public static final String FILTER_2_NAME = "Status";

    public static final String FILTER_2_VALUE = "ACTIVE";

    /*
     * search agreements by two customize filter
     */
    public static void main(String[] args) {

        List<AgreementViewSummary> agreementSummaryList = getAgreements();

        ReferenceCodesUtils.formatOutput(agreementSummaryList);

    }

    public static List<AgreementViewSummary> getAgreements() {
        MarketplaceAgreementClient marketplaceAgreementClient =
            MarketplaceAgreementClient.builder()
                .httpClient(ApacheHttpClient.builder().build())
                .credentialsProvider(ProfileCredentialsProvider.create())
                .build();

        Filter partyTypeFilter = Filter.builder().name(PARTY_TYPE_FILTER_NAME)
            .values(PARTY_TYPE_FILTER_VALUE_PROPOSER).build();

        Filter agreementTypeFilter = Filter.builder().name(AGREEMENT_TYPE_FILTER_NAME)
            .values(AGREEMENT_TYPE_FILTER_VALUE_PURCHASEAGREEMENT).build();

        Filter customizeFilter1 =
            Filter.builder().name(FILTER_1_NAME).values(FILTER_1_VALUE).build();

        Filter customizeFilter2 =
            Filter.builder().name(FILTER_2_NAME).values(FILTER_2_VALUE).build();
```

```
List<Filter> filters = new ArrayList<Filter>();

filters.addAll(Arrays.asList(partyTypeFilter, agreementTypeFilter,
customizeFilter1, customizeFilter2));

SearchAgreementsRequest searchAgreementsRequest =
    SearchAgreementsRequest.builder()
        .catalog(AWS_MP_CATALOG)
        .filters(filters)
        .build();

SearchAgreementsResponse searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

List<AgreementViewSummary> agreementSummaryList = new
ArrayList<AgreementViewSummary>();

agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());

while (searchAgreementsResponse.nextToken() != null &&
searchAgreementsResponse.nextToken().length() > 0) {
    searchAgreementsRequest =
        SearchAgreementsRequest.builder()
            .catalog(AWS_MP_CATALOG)
            .filters(filters)
            .nextToken(searchAgreementsResponse.nextToken())
            .build();
    searchAgreementsResponse =
marketplaceAgreementClient.searchAgreements(searchAgreementsRequest);

    agreementSummaryList.addAll(searchAgreementsResponse.agreementViewSummaries());
}
return agreementSummaryList;
}
}
```

- For API details, see [SearchAgreements](#) in *AWS SDK for Java 2.x API Reference*.

For a complete list of AWS SDK developer guides and code examples, see [Using this service with an AWS SDK](#). This topic also includes information about getting started and details about previous SDK versions.

Actions

The following actions are supported by AWS Marketplace Catalog Service:

- [BatchDescribeEntities](#)
- [CancelChangeSet](#)
- [DeleteResourcePolicy](#)
- [DescribeChangeSet](#)
- [DescribeEntity](#)
- [GetResourcePolicy](#)
- [ListChangeSets](#)
- [ListEntities](#)
- [ListTagsForResource](#)
- [PutResourcePolicy](#)
- [StartChangeSet](#)
- [TagResource](#)
- [UntagResource](#)

The following actions are supported by AWS Marketplace Agreement Service:

- [DescribeAgreement](#)
- [GetAgreementTerms](#)
- [SearchAgreements](#)

The following actions are supported by AWS Marketplace Metering Service:

- [BatchMeterUsage](#)
- [MeterUsage](#)
- [RegisterUsage](#)
- [ResolveCustomer](#)

The following actions are supported by AWS Marketplace Entitlement Service:

- [GetEntitlements](#)

The following actions are supported by AWS Marketplace Deployment Service:

- [ListTagsForResource](#)
- [PutDeploymentParameter](#)
- [TagResource](#)
- [UntagResource](#)

The following actions are supported by AWS Marketplace Reporting Service:

- [GetBuyerDashboard](#)

AWS Marketplace Catalog API

The following actions are supported by AWS Marketplace Catalog API:

- [BatchDescribeEntities](#)
- [CancelChangeSet](#)
- [DeleteResourcePolicy](#)
- [DescribeChangeSet](#)
- [DescribeEntity](#)
- [GetResourcePolicy](#)
- [ListChangeSets](#)
- [ListEntities](#)
- [ListTagsForResource](#)
- [PutResourcePolicy](#)
- [StartChangeSet](#)
- [TagResource](#)
- [UntagResource](#)

BatchDescribeEntities

Service: AWS Marketplace Catalog Service

Returns metadata and content for multiple entities. This is the Batch version of the DescribeEntity API and uses the same IAM permission action as DescribeEntity API.

Request Syntax

```
POST /BatchDescribeEntities HTTP/1.1
Content-type: application/json
```

```
{
  "EntityRequestList": [
    {
      "Catalog": "string",
      "EntityId": "string"
    }
  ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

EntityRequestList

List of entity IDs and the catalogs the entities are present in.

Type: Array of [EntityRequest](#) objects

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "EntityDetails": {
    "string" : {
      "DetailsDocument": JSON value,
      "EntityArn": "string",
      "EntityIdentifier": "string",
      "EntityType": "string",
      "LastModifiedDate": "string"
    }
  },
  "Errors": {
    "string" : {
      "ErrorCode": "string",
      "ErrorMessage": "string"
    }
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

EntityDetails

Details about each entity.

Type: String to [EntityDetail](#) object map

Key Length Constraints: Minimum length of 1. Maximum length of 255.

Key Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Errors

A map of errors returned, with `EntityId` as the key and `errorDetail` as the value.

Type: String to [BatchDescribeErrorDetail](#) object map

Key Length Constraints: Minimum length of 1. Maximum length of 255.

Key Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

Examples

Sample request for SaaS products

You can use `BatchDescribeEntities` to obtain information about multiple software as a service (SaaS) products at the same time. The examples below show how to obtain details on two products at the same time and how the response looks like when one product exists and another one does not.

```
{
```

```

"EntityRequestList": [
  { "EntityId": "example-123-abcd", "Catalog": "AWSMarketplace"},
  { "EntityId": "example-456-abcd", "Catalog": "AWSMarketplace"}
]
}

```

Sample response for SaaS products

This example illustrates one usage of BatchDescribeEntities.

```

{
  "EntityDetails": {
    "example-123-abcd": {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/SaaSProduct/example-123-abcd",
      "EntityType": "SaaSProduct@1.0",
      "EntityIdentifier": "example-123-abcd@1",
      "LastModifiedDate": "2023-07-31T20:37:08Z",
      "DetailsDocument": {
        "Description": {
          "ProductTitle": "Test title",
          "ProductCode": "prod-example123",
          "ShortDescription": "Test description",
          "Manufacturer": "TeamExtension",
          "LongDescription": "The full description.",
          "Sku": null,
          "Highlights": [
            "test highlights"
          ],
          "AssociatedProducts": null,
          "SearchKeywords": [
            "Configuration file management"
          ],
          "Visibility": "Limited",
          "ProductState": "Active",
          "Categories": [
            "Security"
          ]
        },
        "PromotionalResources": {
          "LogoUrl": "http://awsmp-logos.s3.amazonaws.com/11267590-efc3-4be5-b1b2-7e461e1dd2c6_config_Logo.png",
          "Videos": [],
          "AdditionalResources": [

```

```
        {
            "Type": "Link",
            "Text": "Manage your configuration files with ease",
            "Url": "http://www.configapp.com/post/category/tutorials/"
        }
    ],
    "PromotionalMedia": null
},
"SupportInformation": {
    "Description": "Site tutorials, mail and direct contact details",
    "Resources": []
},
"Dimensions": [
    {
        "Name": "Starter Plan",
        "Description": "Starter Plan",
        "Key": "StarterPlan",
        "Unit": "Mbps",
        "Types": [
            "ExternallyMetered"
        ]
    }
],
"Versions": [
    {
        "Id": "version-123",
        "DeliveryOptions": [
            {
                "Id": "do-hg5tfkwp2vxqi",
                "Type": "SoftwareRegistration",
                "FulfillmentUrl": ""
            }
        ]
    }
],
"Targeting": {
    "PositiveTargeting": {
        "BuyerAccounts": [
            "123456789012"
        ]
    }
}
},
}
```

```
    },
    "Errors": {
      "example-456-abcd": {
        "ErrorCode": "ResourceNotFound",
        "ErrorMessage": "Requested entity 'example-456-abcd' does not exist."
      }
    }
  }
}
```

Sample request for offer entities

You can use `BatchDescribeEntities` to obtain information about multiple offers at once. The examples below show how to obtain details on two offers at the same time.

```
{
  "EntityRequestList": [
    { "EntityId": "offer-example123", "Catalog": "AWSMarketplace" },
    { "EntityId": "offer-example456", "Catalog": "AWSMarketplace" }
  ]
}
```

Sample response for offer entities

This example illustrates one usage of `BatchDescribeEntities`.

```
{
  "EntityDetails": {
    "offer-example123": {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Offer/offer-example123",
      "EntityType": "Offer@1.0",
      "EntityIdentifier": "offer-example123@1",
      "LastModifiedDate": "2023-07-31T20:37:08Z",
      "DetailsDocument": {
        "Id": "offer-example123",
        "State": "Released",
        "Name": "Test Offer",
        "Description": "Worldwide offer for Test Product",
        "PreExistingAgreement": {
          "AcquisitionChannel": "External",
          "PricingModel": "Contract"
        },
        "ProductId": "prod-example123",

```

```
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "If you need to request a refund for software sold
by Amazon Web Services, LLC, please contact AWS Customer Service."
      },
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
          }
        ]
      },
      {
        "Type": "FreeTrialPricingTerm",
        "Duration": "P30D",
        "Grants": [
          {
            "DimensionKey": "m3.xlarge",
            "MaxQuantity": 10
          },
          {
            "DimensionKey": "m4.xlarge",
            "MaxQuantity": 10
          }
        ]
      },
      {
        "Type": "ConfigurableUpfrontPricingTerm",
        "CurrencyCode": "USD",
        "RateCards": [
          {
            "Selector": {
              "Type": "Duration",
              "Value": "P365D"
            },
            "RateCard": [
              {
                "DimensionKey": "m3.large",
                "Price": "300.00"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```
        "DimensionKey": "m4.xlarge",
        "Price": "400.00"
    }
  ],
  "Constraints": {
    "MultipleDimensionSelection": "Allowed",
    "QuantityConfiguration": "Allowed"
  }
}
],
},
{
  "Type": "UsageBasedPricingTerm",
  "CurrencyCode": "USD",
  "RateCards": [
    {
      "RateCard": [
        {
          "DimensionKey": "m3.large",
          "Price": "0.10"
        },
        {
          "DimensionKey": "m4.xlarge",
          "Price": "0.20"
        }
      ]
    }
  ]
},
},
{
  "Type": "FixedUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "Price": "200.00",
  "Grants": [
    {
      "DimensionKey": "Users",
      "MaxQuantity": 10
    }
  ]
},
},
{
  "Type": "RecurringPaymentTerm",
  "CurrencyCode": "USD",
  "BillingPeriod": "Monthly",
```

```
        "Price": "100.0"
    },
    {
        "Type": "PaymentScheduleTerm",
        "CurrencyCode": "USD",
        "Schedule": [
            {
                "ChargeDate": "2020-12-01T00:00:00.000Z",
                "ChargeAmount": "1000.00"
            },
            {
                "ChargeDate": "2021-06-15T00:00:00.000Z",
                "ChargeAmount": "1250.00"
            }
        ]
    },
    {
        "Type": "ByolPricingTerm"
    },
    {
        "Type": "RenewalTerm"
    }
],
"Rules": [
    {
        "Type": "TargetingRule",
        "PositiveTargeting": {
            "CountryCodes": [
                "US",
                "CA"
            ],
            "BuyerAccounts": [
                "123456789012"
            ]
        },
        "NegativeTargeting": {
            "CountryCodes": [
                "XX"
            ]
        }
    },
    {
        "Type": "AvailabilityRule",
        "AvailabilityEndDate": "2024-08-30T01:56:03.000Z"
    }
]
```

```
    }
  ]
}
},
"offer-example456": {
  "EntityArn": "arn:aws:aws-marketplace:us-
east-1:123456789012:AWSMarketplace/Offer/offer-example456",
  "EntityType": "Offer@1.0",
  "EntityIdentifier": "offer-example456@1",
  "LastModifiedDate": "2023-07-31T21:37:08Z",
  "DetailsDocument": {
    "Id": "offer-example456",
    "State": "Released",
    "Name": "Test Offer",
    "Description": "Worldwide offer for Test Product",
    "PreExistingAgreement": {
      "AcquisitionChannel": "External",
      "PricingModel": "Contract"
    },
    "ProductId": "prod-example456",
    "Terms": [
      {
        "Type": "SupportTerm",
        "RefundPolicy": "If you need to request a refund for software sold
by Amazon Web Services, LLC, please contact AWS Customer Service."
      },
      {
        "Type": "LegalTerm",
        "Documents": [
          {
            "Type": "CustomEula",
            "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
          }
        ]
      }
    ],
    {
      "Type": "FreeTrialPricingTerm",
      "Duration": "P30D",
      "Grants": [
        {
          "DimensionKey": "m3.xlarge",
          "MaxQuantity": 10
        }
      ]
    }
  ]
}
```



```
        "DimensionKey": "m4.xlarge",
        "MaxQuantity": 10
    }
]
},
{
    "Type": "ConfigurableUpfrontPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
    {
        "Selector": {
            "Type": "Duration",
            "Value": "P365D"
        },
        "RateCard": [
        {
            "DimensionKey": "m3.large",
            "Price": "300.00"
        },
        {
            "DimensionKey": "m4.xlarge",
            "Price": "400.00"
        }
        ],
        "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
        }
    }
]
},
{
    "Type": "UsageBasedPricingTerm",
    "CurrencyCode": "USD",
    "RateCards": [
    {
        "RateCard": [
        {
            "DimensionKey": "m3.large",
            "Price": "0.10"
        },
        {
            "DimensionKey": "m4.xlarge",
            "Price": "0.20"
        }
        ]
    }
]
```

```

        }
      ]
    }
  ]
},
{
  "Type": "FixedUpfrontPricingTerm",
  "CurrencyCode": "USD",
  "Price": "200.00",
  "Grants": [
    {
      "DimensionKey": "Users",
      "MaxQuantity": 10
    }
  ]
},
{
  "Type": "RecurringPaymentTerm",
  "CurrencyCode": "USD",
  "BillingPeriod": "Monthly",
  "Price": "100.0"
},
{
  "Type": "PaymentScheduleTerm",
  "CurrencyCode": "USD",
  "Schedule": [
    {
      "ChargeDate": "2020-12-01T00:00:00.000Z",
      "ChargeAmount": "1000.00"
    },
    {
      "ChargeDate": "2021-06-15T00:00:00.000Z",
      "ChargeAmount": "1250.00"
    }
  ]
},
{
  "Type": "ByolPricingTerm"
},
{
  "Type": "RenewalTerm"
}
],
"Rules": [

```

```
    {
      "Type": "TargetingRule",
      "PositiveTargeting": {
        "CountryCodes": [
          "US",
          "CA"
        ],
        "BuyerAccounts": [
          "123456789012"
        ]
      },
      "NegativeTargeting": {
        "CountryCodes": [
          "XX"
        ]
      }
    },
    {
      "Type": "AvailabilityRule",
      "AvailabilityEndDate": "2024-08-30T01:56:03.000Z"
    }
  ]
},
}
```

Sample request for resale authorization entities

You can use `BatchDescribeEntities` to obtain information on multiple resale authorizations at once. The examples below show how to obtain details on two resale authorizations at the same time.

```
{
  "EntityRequestList": [
    { "EntityId": "resaleauthz-example123", "Catalog": "AWSMarketplace" },
    { "EntityId": "resaleauthz-example456", "Catalog": "AWSMarketplace" }
  ]
}
```

Sample response for resale authorization entities

This example illustrates one usage of BatchDescribeEntities.

```
{
  "EntityDetails": {
    "resaleauthz-example123": {
      "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ResaleAuthorization/resaleauthz-example123",
      "EntityType": "ResaleAuthorization@1.0",
      "EntityIdentifier": "resaleauthz-example123@1",
      "LastModifiedDate": "2023-07-31T20:37:08Z",
      "DetailsDocument": {
        "Name": "TestResaleAuthorization",
        "Description": "ResaleAuthorization for Test Product",
        "ProductId": "prod-example123",
        "ProductName": "TestProduct",
        "Status": "Active",
        "PreExistingBuyerAgreement": {
          "AcquisitionChannel": "Unknown",
          "PricingModel": "Unknown"
        },
        "CreatedDate": "2023-07-18T16:39:31.335Z",
        "ManufacturerLegalName": "ChannelCAPI.Inc",
        "ManufacturerAccountId": "123456789012",
        "Dimensions": [
          {
            "Name": "Protected Resources",
            "Description": "Additional 100 protected resources",
            "Key": "hundredresources",
            "Unit": "Units",
            "Types": [
              "Entitled"
            ]
          }
        ],
        "OfferDetails": {
          "OfferExtendedStatus": "Not Started",
          "OfferCreatedCount": 0
        },
        "Terms": [
          {
            "Type": "ResaleUsageBasedPricingTerm",
            "Id": "term_id_placeholder",

```

```
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "RateCard": [
          {
            "DimensionKey": "resource_number",
            "Price": "0.05"
          },
          {
            "DimensionKey": "scanned_data",
            "Price": "0.05"
          }
        ]
      }
    ],
  },
  {
    "Type": "ResaleConfigurableUpfrontPricingTerm",
    "Id": "term_id_placeholder",
    "CurrencyCode": "USD",
    "RateCards": [
      {
        "Selector": {
          "Type": "Duration",
          "Value": "P24M"
        },
        "RateCard": [
          {
            "DimensionKey": "hundredresources",
            "Price": "0.04"
          },
          {
            "DimensionKey": "tenTBData",
            "Price": "0.03"
          },
          {
            "DimensionKey": "channel_custom",
            "Price": "0.02"
          }
        ],
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        }
      }
    ]
  }
}
```

```
    }
  ]
},
{
  "Type": "ResaleFixedUpfrontPricingTerm",
  "Id": "term-123",
  "CurrencyCode": "USD",
  "Duration": "P180D",
  "Price": "0.0",
  "Grants": [
    {
      "DimensionKey": "example1230xm1",
      "MaxQuantity": 1
    }
  ]
},
{
  "Type": "ResalePaymentScheduleTerm",
  "Id": "term-123",
  "CurrencyCode": "USD",
  "Schedule": [
    {
      "ChargeDate": "2018-07-01T00:00:00.000Z",
      "ChargeAmount": "200.00"
    },
    {
      "ChargeDate": "2019-05-01T00:00:00.000Z",
      "ChargeAmount": "200.00"
    }
  ]
},
{
  "Type": "BuyerLegalTerm",
  "Id": "term_id_placeholder",
  "Documents": [
    {
      "Type": "StandardEula",
      "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
    }
  ]
},
{
  "Type": "ResaleLegalTerm",
  "Id": "term_id_placeholder",
```

```
    "Documents": [
      {
        "Type": "StandardResellerContract",
        "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
      }
    ],
    {
      "Type": "BuyerValidityTerm",
      "Id": "term_id_placeholder",
      "MaximumAgreementStartDate": "2023-09-25T23:59:59.000Z"
    },
    {
      "Type": "BuyerTargetingTerm",
      "Id": "term_id_placeholder",
      "PositiveTargeting": {
        "BuyerAccounts": [
          {
            "AwsAccountId": "123456789012"
          }
        ]
      }
    }
  ],
  "Rules": [
    {
      "Type": "AvailabilityRule",
      "Id": "availability_rule_id_placeholder",
      "Usage": "Limited",
      "AvailabilityEndDate": "2024-05-31T23:59:59Z",
      "OffersMaxQuantity": 1
    },
    {
      "Type": "PartnerTargetingRule",
      "Id": "partner_targeting_rule_id_placeholder",
      "ResellerAccountId": "123456789012",
      "ResellerLegalName": "ChannelCAPICP.Inc"
    }
  ]
},
"resaleauthz-example456": {
  "EntityArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ResaleAuthorization/resaleauthz-example456",
```

```
"EntityType": "ResaleAuthorization@1.0",
"EntityIdentifier": "resaleauthz-example456@1",
"LastModifiedDate": "2023-07-31T22:37:08Z",
"DetailsDocument": {
  "Name": "TestResaleAuthorization",
  "Description": "ResaleAuthorization for Test Product",
  "ProductId": "prod-example456",
  "ProductName": "TestProduct",
  "Status": "Active",
  "PreExistingBuyerAgreement": {
    "AcquisitionChannel": "Unknown",
    "PricingModel": "Unknown"
  },
  "CreatedDate": "2023-07-18T16:39:31.335Z",
  "ManufacturerLegalName": "ChannelCAPI.Inc",
  "ManufacturerAccountId": "123456789012",
  "Dimensions": [
    {
      "Name": "Protected Resources",
      "Description": "Additional 100 protected resources",
      "Key": "hundredresources",
      "Unit": "Units",
      "Types": [
        "Entitled"
      ]
    }
  ],
  "OfferDetails": {
    "OfferExtendedStatus": "Not Started",
    "OfferCreatedCount": 0
  },
  "Terms": [
    {
      "Type": "ResaleUsageBasedPricingTerm",
      "Id": "term_id_placeholder",
      "CurrencyCode": "USD",
      "RateCards": [
        {
          "RateCard": [
            {
              "DimensionKey": "resource_number",
              "Price": "0.05"
            }
          ]
        }
      ]
    }
  ]
}
```



```

        "DimensionKey": "scanned_data",
        "Price": "0.05"
    }
    ]
}
],
},
{
    "Type": "ResaleConfigurableUpfrontPricingTerm",
    "Id": "term_id_placeholder",
    "CurrencyCode": "USD",
    "RateCards": [
    {
        "Selector": {
            "Type": "Duration",
            "Value": "P24M"
        },
        "RateCard": [
        {
            "DimensionKey": "hundredresources",
            "Price": "0.04"
        },
        {
            "DimensionKey": "tenTBData",
            "Price": "0.03"
        },
        {
            "DimensionKey": "channel_custom",
            "Price": "0.02"
        }
        ],
        "Constraints": {
            "MultipleDimensionSelection": "Allowed",
            "QuantityConfiguration": "Allowed"
        }
    }
    ]
},
{
    "Type": "ResaleFixedUpfrontPricingTerm",
    "Id": "term-123",
    "CurrencyCode": "USD",
    "Duration": "P180D",
    "Price": "0.0",

```

```
    "Grants": [
      {
        "DimensionKey": "example1230xm1",
        "MaxQuantity": 1
      }
    ],
  },
  {
    "Type": "ResalePaymentScheduleTerm",
    "Id": "term-123",
    "CurrencyCode": "USD",
    "Schedule": [
      {
        "ChargeDate": "2018-07-01T00:00:00.000Z",
        "ChargeAmount": "200.00"
      },
      {
        "ChargeDate": "2019-05-01T00:00:00.000Z",
        "ChargeAmount": "200.00"
      }
    ]
  },
  {
    "Type": "BuyerLegalTerm",
    "Id": "term_id_placeholder",
    "Documents": [
      {
        "Type": "StandardEula",
        "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
      }
    ]
  },
  {
    "Type": "ResaleLegalTerm",
    "Id": "term_id_placeholder",
    "Documents": [
      {
        "Type": "StandardResellerContract",
        "Url": "https://s3.amazonaws.com/EULA/custom-eula-1234.txt"
      }
    ]
  },
  {
    "Type": "BuyerValidityTerm",
```

```
        "Id": "term_id_placeholder",
        "MaximumAgreementStartDate": "2023-09-25T23:59:59.000Z"
    },
    {
        "Type": "BuyerTargetingTerm",
        "Id": "term_id_placeholder",
        "PositiveTargeting": {
            "BuyerAccounts": [
                {
                    "AwsAccountId": "123456789012"
                }
            ]
        }
    }
],
"Rules": [
    {
        "Type": "AvailabilityRule",
        "Id": "availability_rule_id_placeholder",
        "Usage": "Limited",
        "AvailabilityEndDate": "2024-05-31T23:59:59Z",
        "OffersMaxQuantity": 1
    },
    {
        "Type": "PartnerTargetingRule",
        "Id": "partner_targeting_rule_id_placeholder",
        "ResellerAccountId": "123456789012",
        "ResellerLegalName": "ChannelCAPICP.Inc"
    }
]
}
}
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CancelChangeSet

Service: AWS Marketplace Catalog Service

Used to cancel an open change request. Must be sent before the status of the request changes to APPLYING, the final stage of completing your change request. You can describe a change during the 60-day request history retention period for API calls.

Request Syntax

```
PATCH /CancelChangeSet?catalog=Catalog&changeSetId=ChangeSetId HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

Catalog

Required. The catalog related to the request. Fixed value: `AWSMarketplace`.

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

ChangeSetId

Required. The unique identifier of the `StartChangeSet` request that you want to cancel.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "ChangeSetArn": "string",
  "ChangeSetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ChangeSetArn

The ARN associated with the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

ChangeSetId

The unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceInUseException

The resource is currently in use.

HTTP Status Code: 423

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteResourcePolicy

Service: AWS Marketplace Catalog Service

Deletes a resource-based policy on an entity that is identified by its resource ARN.

Request Syntax

```
DELETE /DeleteResourcePolicy?resourceArn=ResourceArn HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

ResourceArn

The Amazon Resource Name (ARN) of the entity resource that is associated with the resource policy.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=/, .@-]+(\/[\w+=/, .@-]+)*$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeChangeSet

Service: AWS Marketplace Catalog Service

Provides information about a given change set.

Request Syntax

```
GET /DescribeChangeSet?catalog=Catalog&changeSetId=ChangeSetId HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

Catalog

Required. The catalog related to the request. Fixed value: `AWSMarketplace`

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

ChangeSetId

Required. The unique identifier for the `StartChangeSet` request that you want to describe the details for.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[w\-\]+$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "ChangeSet": [
    {
      "ChangeName": "string",
      "ChangeType": "string",
      "Details": "string",
      "DetailsDocument": JSON value,
      "Entity": {
        "Identifier": "string",
        "Type": "string"
      },
      "ErrorDetailList": [
        {
          "ErrorCode": "string",
          "ErrorMessage": "string"
        }
      ]
    }
  ],
  "ChangeSetArn": "string",
  "ChangeSetId": "string",
  "ChangeSetName": "string",
  "EndTime": "string",
  "FailureCode": "string",
  "FailureDescription": "string",
  "Intent": "string",
  "StartTime": "string",
  "Status": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ChangeSet

An array of [ChangeSummary](#) objects.

Type: Array of [ChangeSummary](#) objects

ChangeSetArn

The ARN associated with the unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

ChangeSetId

Required. The unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

ChangeSetName

The optional name provided in the `StartChangeSet` request. If you do not provide a name, one is set by default.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^[\\w\\s+=.:@-]+$`

EndTime

The date and time, in ISO 8601 format (2018-02-27T13:45:22Z), the request transitioned to a terminal state. The change cannot transition to a different state. Null if the request is not in a terminal state.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\\d]{4})\\-(1[0-2]|0[1-9])\\-(3[01]|0[1-9]|[12][\\d])T(2[0-3]|[01][\\d]):([0-5][\\d]):([0-5][\\d])Z$`

FailureCode

Returned if the change set is in FAILED status. Can be either CLIENT_ERROR, which means that there are issues with the request (see the `ErrorDetailList`), or SERVER_FAULT, which means that there is a problem in the system, and you should retry your request.

Type: String

Valid Values: CLIENT_ERROR | SERVER_FAULT

FailureDescription

Returned if there is a failure on the change set, but that failure is not related to any of the changes in the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^(.)+$`

Intent

The optional intent provided in the `StartChangeSet` request. If you do not provide an intent, APPLY is set by default.

Type: String

Valid Values: VALIDATE | APPLY

StartTime

The date and time, in ISO 8601 format (2018-02-27T13:45:22Z), the request started.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Status

The status of the change request.

Type: String

Valid Values: PREPARING | APPLYING | SUCCEEDED | CANCELLED | FAILED

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeEntity

Service: AWS Marketplace Catalog Service

Returns the metadata and content of the entity.

Request Syntax

```
GET /DescribeEntity?catalog=Catalog&entityId=EntityId HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

Catalog

Required. The catalog related to the request. Fixed value: `AWSMarketplace`

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

EntityId

Required. The unique ID of the entity to describe.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200  
Content-type: application/json
```

```
{
  "Details": "string",
  "DetailsDocument": JSON value,
  "EntityArn": "string",
  "EntityIdentifier": "string",
  "EntityType": "string",
  "LastModifiedDate": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Details

This stringified JSON object includes the details of the entity.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 16384.

Pattern: `^\s*\{[\s\S]*\}\s*$`

DetailsDocument

The JSON value of the details specific to the entity.

To download "DetailsDocument" shapes, see the [Python](#) and [Java](#) shapes on GitHub.

Type: JSON value

EntityArn

The ARN associated to the unique identifier for the entity referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

EntityIdentifier

The identifier of the entity, in the format of EntityId@RevisionId.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\w\-\@]+\$\`

EntityType

The named type of the entity, in the format of `EntityType@Version`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+\$\`

LastModifiedDate

The last modified date of the entity, in ISO 8601 format (2018-02-27T13:45:22Z).

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):(0-5)[\d]):(0-5)[\d])Z$\`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ResourceNotSupportedException

Currently, the specified resource is not supported.

HTTP Status Code: 415

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetResourcePolicy

Service: AWS Marketplace Catalog Service

Gets a resource-based policy of an entity that is identified by its resource ARN.

Request Syntax

```
GET /GetResourcePolicy?resourceArn=ResourceArn HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

ResourceArn

The Amazon Resource Name (ARN) of the entity resource that is associated with the resource policy.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(/[\w+=, .@-]+)*$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "Policy": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Policy

The policy document to set; formatted in JSON.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 10240.

Pattern: `^\[\u0009\u000A\u000D\u0020-\u00FF]+\`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListChangeSets

Service: AWS Marketplace Catalog Service

Returns the list of change sets owned by the account being used to make the call. You can filter this list by providing any combination of `entityId`, `ChangeSetName`, and `status`. If you provide more than one filter, the API operation applies a logical AND between the filters.

You can describe a change during the 60-day request history retention period for API calls.

Request Syntax

```
POST /ListChangeSets HTTP/1.1
Content-type: application/json

{
  "Catalog": "string",
  "FilterList": [
    {
      "Name": "string",
      "ValueList": [ "string" ]
    }
  ],
  "MaxResults": number,
  "NextToken": "string",
  "Sort": {
    "SortBy": "string",
    "SortOrder": "string"
  }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Catalog

The catalog related to the request. Fixed value: `AWSMarketplace`

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

FilterList

An array of filter objects.

Type: Array of [Filter](#) objects

Array Members: Minimum number of 1 item. Maximum number of 8 items.

Required: No

MaxResults

The maximum number of results returned by a single call. This value must be provided in the next call to retrieve the next set of results. By default, this value is 20.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 20.

Required: No

NextToken

The token value retrieved from a previous call to access the next page of results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[\w+=.:@\-\|/]+$`

Required: No

Sort

An object that contains two attributes, `SortBy` and `SortOrder`.

Type: [Sort](#) object

Required: No

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "ChangeSetSummaryList": [
    {
      "ChangeSetArn": "string",
      "ChangeSetId": "string",
      "ChangeSetName": "string",
      "EndTime": "string",
      "EntityIdList": [ "string" ],
      "FailureCode": "string",
      "StartTime": "string",
      "Status": "string"
    }
  ],
  "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ChangeSetSummaryList

Array of `ChangeSetSummaryListItem` objects.

Type: Array of [ChangeSetSummaryListItem](#) objects

NextToken

The value of the next token, if it exists. Null if there are no more results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^\[\w+=.:\@-\|\/\]$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListEntities

Service: AWS Marketplace Catalog Service

Provides the list of entities of a given type.

Request Syntax

```
POST /ListEntities HTTP/1.1
Content-type: application/json

{
  "Catalog": "string",
  "EntityType": "string",
  "EntityTypeFilters": { ... },
  "EntityTypeSort": { ... },
  "FilterList": [
    {
      "Name": "string",
      "ValueList": [ "string" ]
    }
  ],
  "MaxResults": number,
  "NextToken": "string",
  "OwnershipType": "string",
  "Sort": {
    "SortBy": "string",
    "SortOrder": "string"
  }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Catalog

The catalog related to the request. Fixed value: `AWSMarketplace`

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

EntityType

The type of entities to retrieve. Valid values are: `AmiProduct`, `ContainerProduct`, `DataProduct`, `SaaSProduct`, `ProcurementPolicy`, `Experience`, `Audience`, `BrandingSettings`, `Offer`, `Seller`, `ResaleAuthorization`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: Yes

EntityTypeFilters

A Union object containing filter shapes for all `EntityType`s. Each `EntityTypeFilter` shape will have filters applicable for that `EntityType` that can be used to search or filter entities.

Type: [EntityTypeFilters](#) object

Note: This object is a Union. Only one member of this object can be specified or returned.

Required: No

EntityTypeSort

A Union object containing `Sort` shapes for all `EntityType`s. Each `EntityTypeSort` shape will have `SortBy` and `SortOrder` applicable for fields on that `EntityType`. This can be used to sort the results of the filter query.

Type: [EntityTypeSort](#) object

Note: This object is a Union. Only one member of this object can be specified or returned.

Required: No

FilterList

An array of filter objects. Each filter object contains two attributes, `filterName` and `filterValues`.

Type: Array of [Filter](#) objects

Array Members: Minimum number of 1 item. Maximum number of 8 items.

Required: No

[MaxResults](#)

Specifies the upper limit of the elements on a single page. If a value isn't provided, the default value is 20.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

[NextToken](#)

The value of the next token, if it exists. Null if there are no more results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[\w+=.:@\-\|/]*$`

Required: No

[OwnershipType](#)

Filters the returned set of entities based on their owner. The default is SELF. To list entities shared with you through AWS Resource Access Manager (AWS RAM), set to SHARED. Entities shared through the AWS Marketplace Catalog API `PutResourcePolicy` operation can't be discovered through the SHARED parameter.

Type: String

Valid Values: SELF | SHARED

Required: No

[Sort](#)

An object that contains two attributes, `SortBy` and `SortOrder`.

Type: [Sort](#) object

Required: No

Response Syntax

HTTP/1.1 200

Content-type: application/json

```
{
  "EntitySummaryList": [
    {
      "AmiProductSummary": {
        "ProductTitle": "string",
        "Visibility": "string"
      },
      "ContainerProductSummary": {
        "ProductTitle": "string",
        "Visibility": "string"
      },
      "DataProductSummary": {
        "ProductTitle": "string",
        "Visibility": "string"
      },
      "EntityArn": "string",
      "EntityId": "string",
      "EntityType": "string",
      "LastModifiedDate": "string",
      "Name": "string",
      "OfferSummary": {
        "AvailabilityEndDate": "string",
        "BuyerAccounts": [ "string" ],
        "Name": "string",
        "ProductId": "string",
        "ReleaseDate": "string",
        "ResaleAuthorizationId": "string",
        "State": "string",
        "Targeting": [ "string" ]
      },
      "ResaleAuthorizationSummary": {
        "AvailabilityEndDate": "string",
        "CreatedDate": "string",
        "ManufacturerAccountId": "string",

```

```

    "ManufacturerLegalName": "string",
    "Name": "string",
    "OfferExtendedStatus": "string",
    "ProductId": "string",
    "ProductName": "string",
    "ResellerAccountID": "string",
    "ResellerLegalName": "string",
    "Status": "string"
  },
  "SaaSProductSummary": {
    "ProductTitle": "string",
    "Visibility": "string"
  },
  "Visibility": "string"
}
],
"NextToken": "string"
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

EntitySummaryList

Array of EntitySummary objects.

Type: Array of [EntitySummary](#) objects

NextToken

The value of the next token if it exists. Null if there is no more result.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[\\w+=.:@\\-\\/]$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListTagsForResource

Service: AWS Marketplace Catalog Service

Lists all tags that have been added to a resource (either an [entity](#) or [change set](#)).

Request Syntax

```
POST /ListTagsForResource HTTP/1.1
Content-type: application/json
```

```
{
  "ResourceArn": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

[ResourceArn](#)

Required. The Amazon Resource Name (ARN) associated with the resource you want to list tags on.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(\/[\w+=, .@-]+)*$`

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "ResourceArn": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ResourceArn

Required. The ARN associated with the resource you want to list tags on.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(/[\w+=, .@-]+)*$`

Tags

Required. A list of objects specifying each key name and value. Number of objects allowed: 1-50.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServerErrorException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

PutResourcePolicy

Service: AWS Marketplace Catalog Service

Attaches a resource-based policy to an entity. Examples of an entity include: AmiProduct and ContainerProduct.

Request Syntax

```
POST /PutResourcePolicy HTTP/1.1
Content-type: application/json
```

```
{
  "Policy": "string",
  "ResourceArn": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Policy

The policy document to set; formatted in JSON.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 10240.

Pattern: `^[\u0009\u000A\u000D\u0020-\u00FF]+$`

Required: Yes

ResourceArn

The Amazon Resource Name (ARN) of the entity resource you want to associate with a resource policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(#[\w+=, .@-]+)*$`

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServerError

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartChangeSet

Service: AWS Marketplace Catalog Service

Allows you to request changes for your entities. Within a single ChangeSet, you can't start the same change type against the same entity multiple times. Additionally, when a ChangeSet is running, all the entities targeted by the different changes are locked until the change set has completed (either succeeded, cancelled, or failed). If you try to start a change set containing a change against an entity that is already locked, you will receive a `ResourceInUseException` error.

For example, you can't start the ChangeSet described in the [example](#) later in this topic because it contains two changes to run the same change type (`AddRevisions`) against the same entity (`entity-id@1`).

For more information about working with change sets, see [Working with change sets](#). For information about change types for single-AMI products, see [Working with single-AMI products](#). Also, for more information about change types available for container-based products, see [Working with container products](#).

To download "DetailsDocument" shapes, see [Python](#) and [Java](#) shapes on GitHub.

Request Syntax

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "string",
  "ChangeSet": [
    {
      "ChangeName": "string",
      "ChangeType": "string",
      "Details": "string",
      "DetailsDocument": JSON value,
      "Entity": {
        "Identifier": "string",
        "Type": "string"
      },
      "EntityTags": [
        {
          "Key": "string",
```

```
        "Value": "string"
      }
    ]
  },
  "ChangeSetName": "string",
  "ChangeSetTags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "ClientRequestToken": "string",
  "Intent": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Catalog

The catalog related to the request. Fixed value: AWSMarketplace

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

ChangeSet

Array of change object.

Type: Array of [Change](#) objects

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Required: Yes

ChangeSetName

Optional case sensitive string of up to 100 ASCII characters. The change set name can be used to filter the list of change sets.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^[\\w\\s+=.:@-]+$`

Required: No

ChangeSetTags

A list of objects specifying each key name and value for the ChangeSetTags property.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Required: No

ClientRequestToken

A unique token to identify the request to ensure idempotency.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[!-~]+$`

Required: No

Intent

The intent related to the request. The default is APPLY. To test your request before applying changes to your entities, use VALIDATE. This feature is currently available for adding versions to single-AMI products. For more information, see [Add a new version](#).

Type: String

Valid Values: VALIDATE | APPLY

Required: No

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "ChangeSetArn": "string",
  "ChangeSetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ChangeSetArn

The ARN associated to the unique identifier generated for the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

ChangeSetId

Unique identifier generated for the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[w\ -]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceInUseException

The resource is currently in use.

HTTP Status Code: 423

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ServiceQuotaExceededException

The maximum number of open requests per account has been exceeded.

HTTP Status Code: 402

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

Examples

Example

You can't start this `ChangeSet` because it contains two changes to run the same change type (`AddRevisions`) against the same entity (`entity-id@1`).

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSetName": "Adding revisions to my test Data Product",
  "ChangeSet": [
    {
      "ChangeType": "AddRevisions",
      "Entity": {
        "Identifier": "entity-id@1",
        "Type": "DataProduct@1.0"
      },
      "Details": "{\"DataSetArn\": \"data-set-arn\", \"RevisionArns\": [\"revision-arn\", \"revision-arn-2\"] }"
    },
    {
      "ChangeType": "AddRevisions",
      "Entity": {
        "Identifier": "entity-id@1",
        "Type": "DataProduct@1.0"
      },
      "Details": "{\"DataSetArn\": \"data-set-arn\", \"RevisionArns\": [\"revision-arn3\"] }"
    }
  ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

TagResource

Service: AWS Marketplace Catalog Service

Tags a resource (either an [entity](#) or [change set](#)).

Request Syntax

```
POST /TagResource HTTP/1.1
Content-type: application/json

{
  "ResourceArn": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

[ResourceArn](#)

Required. The Amazon Resource Name (ARN) associated with the resource you want to tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(\/[\w+=, .@-]+)*$`

Required: Yes

Tags

Required. A list of objects specifying each key name and value. Number of objects allowed: 1-50.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UntagResource

Service: AWS Marketplace Catalog Service

Removes a tag or list of tags from a resource (either an [entity](#) or [change set](#)).

Request Syntax

```
POST /UntagResource HTTP/1.1
Content-type: application/json

{
  "ResourceArn": "string",
  "TagKeys": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

[ResourceArn](#)

Required. The Amazon Resource Name (ARN) associated with the resource you want to remove the tag from.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^arn:[\w+=/, .@-]+:aws-marketplace:[\w+=/, .@-]*:[0-9]+:[\w+=, .@-]+(\/[\w+=, .@-]+)*$`

Required: Yes

[TagKeys](#)

Required. A list of key names of tags to be removed. Number of strings allowed: 0-256.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: $^([\p{L}\p{Z}\p{N}_\cdot :/=+\-@]^*)\$$

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Agreement API

The following actions are supported by AWS Marketplace Agreement API:

- [DescribeAgreement](#)
- [GetAgreementTerms](#)
- [SearchAgreements](#)

DescribeAgreement

Service: AWS Marketplace Agreement Service

Provides details about an agreement, such as the proposer, acceptor, start date, and end date.

Request Syntax

```
{  
  "agreementId": "string"  
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

agreementId

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: Yes

Response Syntax

```
{  
  "acceptanceTime": number,  
  "acceptor": {  
    "accountId": "string"  
  },  
  "agreementId": "string",
```

```
"agreementType": "string",
"endTime": number,
"estimatedCharges": {
  "agreementValue": "string",
  "currencyCode": "string"
},
"proposalSummary": {
  "offerId": "string",
  "resources": [
    {
      "id": "string",
      "type": "string"
    }
  ]
},
"proposer": {
  "accountId": "string"
},
"startTime": number,
"status": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

acceptanceTime

The date and time the offer was accepted or the agreement was created.

Note

AcceptanceTime and StartTime can differ for future dated agreements (FDAs).

Type: Timestamp

acceptor

The details of the party accepting the agreement terms. This is commonly the buyer for PurchaseAgreement.

Type: [Acceptor](#) object

[agreementId](#)

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

[agreementType](#)

The type of agreement. Values are `PurchaseAgreement` or `VendorInsightsAgreement`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z]+$`

[endTime](#)

The date and time when the agreement ends. The field is `null` for pay-as-you-go agreements, which don't have end dates.

Type: Timestamp

[estimatedCharges](#)

The estimated cost of the agreement.

Type: [EstimatedCharges](#) object

[proposalSummary](#)

A summary of the proposal received from the proposer.

Type: [ProposalSummary](#) object

[proposer](#)

The details of the party proposing the agreement terms. This is commonly the seller for `PurchaseAgreement`.

Type: [Proposer](#) object

startTime

The date and time when the agreement starts.

Type: Timestamp

status

The current status of the agreement.

Statuses include:

- ACTIVE – The terms of the agreement are active.
- ARCHIVED – The agreement ended without a specified reason.
- CANCELLED – The acceptor ended the agreement before the defined end date.
- EXPIRED – The agreement ended on the defined end date.
- RENEWED – The agreement was renewed into a new agreement (for example, an auto-renewal).
- REPLACED – The agreement was replaced using an agreement replacement offer.
- TERMINATED – The agreement ended before the defined end date because of an AWS termination (for example, a payment failure).

Type: String

Valid Values: ACTIVE | ARCHIVED | CANCELLED | EXPIRED | RENEWED | REPLACED | TERMINATED

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerErrorException

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 400

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 400

ValidationException

The input fails to satisfy the constraints specified by the service.

HTTP Status Code: 400

Examples

Sample request

This example illustrates one usage of DescribeAgreement.

```
{
  "agreementId" : "fEXAMPLE-0aa6-4e42-8715-6a1EXAMPLE95"
}
```

Sample response

This example illustrates one usage of DescribeAgreement.

```
{
  "agreementId": "fEXAMPLE-0aa6-4e42-8715-6a1EXAMPLE95",
  "acceptor": {
    "accountId": "123456789010"
  },
  "proposer": {
    "accountId": "123456789010"
  },
  "startTime": 2019-10-08T21:40:43.644Z,
  "endTime": 2023-10-08T21:40:43.644Z,
  "acceptanceTime": 2019-10-08T00:00:00.000Z,
```

```
"agreementType": "PurchaseAgreement",
"proposalSummary": {
  "resources": [
    {
      "id": "0EXAMPLE-8ce8-4814-bcf1-636EXAMPLEb5",
      "type": "AmiProduct"
    }
  ],
  "offerId": "ABCDEFGHIJKLMN123"
},
"status": "ACTIVE",
"estimatedCharges": {
  "currencyCode": "USD",
  "agreementValue": "1000"
}
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetAgreementTerms

Service: AWS Marketplace Agreement Service

Obtains details about the terms in an agreement that you participated in as proposer or acceptor.

The details include:

- **TermType** – The type of term, such as `LegalTerm`, `RenewalTerm`, or `ConfigurableUpfrontPricingTerm`.
- **TermID** – The ID of the particular term, which is common between offer and agreement.
- **TermPayload** – The key information contained in the term, such as the EULA for `LegalTerm` or pricing and dimensions for various pricing terms, such as `ConfigurableUpfrontPricingTerm` or `UsageBasedPricingTerm`.
- **Configuration** – The buyer/acceptor's selection at the time of agreement creation, such as the number of units purchased for a dimension or setting the `EnableAutoRenew` flag.

Request Syntax

```
{
  "agreementId": "string",
  "maxResults": number,
  "nextToken": "string"
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

agreementId

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: Yes

maxResults

The maximum number of agreements to return in the response.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

nextToken

A token to specify where to start pagination

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: `^[a-zA-Z0-9+/=]+$`

Required: No

Response Syntax

```
{
  "acceptedTerms": [
    { ... }
  ],
  "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

acceptedTerms

A subset of terms proposed by the proposer that have been accepted by the acceptor as part of the agreement creation.

Type: Array of [AcceptedTerm](#) objects

nextToken

A token to specify where to start pagination

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: `^[a-zA-Z0-9+/=]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerError

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 400

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 400

ValidationException

The input fails to satisfy the constraints specified by the service.

HTTP Status Code: 400

Examples

Sample request

This example illustrates one usage of `GetAgreementTerms`.

```
{
  "agreementId": "fEXAMPLE-0aa6-4e42-8715-6a1EXAMPLE95"
}
```

Sample response

This example illustrates one usage of `GetAgreementTerms`.

```
{
  "AcceptedTerms": [{
    "ConfigurableUpfrontPricingTerm": {
      "Type": "ConfigurableUpfrontPricingTerm",
      "CurrencyCode": "USD",
      "RateCards": [{
        "Selector": {
          "Type": "Duration",
          "Value": "P24M"
        },
        "Constraints": {
          "MultipleDimensionSelection": "Allowed",
          "QuantityConfiguration": "Allowed"
        },
        "RateCard": [{
          "DimensionKey": "AdminUsers",
          "Price": "0.5"
        },
        {
          "DimensionKey": "ReadOnlyUsers",
          "Price": "1"
        }
      ]
    },
    {
      "Selector": {
        "Type": "Duration",
```

```
        "Value": "P36M"
      },
      "Constraints": {
        "MultipleDimensionSelection": "Allowed",
        "QuantityConfiguration": "Allowed"
      },
      "RateCard": [{
        "DimensionKey": "AdminUsers",
        "Price": "1"
      },
      {
        "DimensionKey": "ReadOnlyUsers",
        "Price": "2"
      }
    ]
  },
  "Configuration": {
    "Dimensions": [{
      "DimensionKey": "AdminUsers",
      "DimensionValue": 1
    }],
    "SelectorValue": "P24M"
  }
},
{
  "RenewalTerm": {
    "Type": "RenewalTerm",
    "Configuration": {
      "EnableAutoRenew": false
    }
  }
},
{
  "legalTerm": {
    "type": "LegalTerm",
    "documents": [{
      "type": "CustomEula",
      "url": "URL"
    }
  ]
}
}
```

```
  ],  
  "nextToken": null  
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

SearchAgreements

Service: AWS Marketplace Agreement Service

Searches across all agreements that a proposer or an acceptor has in AWS Marketplace. The search returns a list of agreements with basic agreement information.

The following filter combinations are supported:

- PartyType as Proposer + AgreementType + ResourceIdentifier
- PartyType as Proposer + AgreementType + OfferId
- PartyType as Proposer + AgreementType + AcceptorAccountId
- PartyType as Proposer + AgreementType + Status
- PartyType as Proposer + AgreementType + ResourceIdentifier + Status
- PartyType as Proposer + AgreementType + OfferId + Status
- PartyType as Proposer + AgreementType + AcceptorAccountId + Status
- PartyType as Proposer + AgreementType + ResourceType + Status
- PartyType as Proposer + AgreementType + AcceptorAccountId + ResourceType + Status
- PartyType as Proposer + AgreementType + AcceptorAccountId + OfferId
- PartyType as Proposer + AgreementType + AcceptorAccountId + OfferId + Status
- PartyType as Proposer + AgreementType + AcceptorAccountId + ResourceIdentifier
- PartyType as Proposer + AgreementType + AcceptorAccountId + ResourceIdentifier + Status
- PartyType as Proposer + AgreementType + AcceptorAccountId + ResourceType

Request Syntax

```
{
  "catalog": "string",
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
}
```

```
"maxResults": number,  
"nextToken": "string",  
"sort": {  
  "sortBy": "string",  
  "sortOrder": "string"  
}  
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

catalog

The catalog in which the agreement was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: No

filters

The filter name and value pair used to return a specific list of results.

The following filters are supported:

- `ResourceIdentifier` – The unique identifier of the resource.
- `ResourceType` – Type of the resource, which is the product (`AmiProduct`, `ContainerProduct`, or `SaaSProduct`).
- `PartyType` – The party type (either `Acceptor` or `Proposer`) of the caller. For agreements where the caller is the proposer, use the `Proposer` filter. For agreements where the caller is the acceptor, use the `Acceptor` filter.

- `AccepterAccountId` – The AWS account ID of the party accepting the agreement terms.
- `OfferId` – The unique identifier of the offer in which the terms are registered in the agreement token.
- `Status` – The current status of the agreement. Values include `ACTIVE`, `ARCHIVED`, `CANCELLED`, `EXPIRED`, `RENEWED`, `REPLACED`, and `TERMINATED`.
- `BeforeEndTime` – A date used to filter agreements with a date before the `endTime` of an agreement.
- `AfterEndTime` – A date used to filter agreements with a date after the `endTime` of an agreement.
- `AgreementType` – The type of agreement. Values include `PurchaseAgreement` or `VendorInsightsAgreement`.

Type: Array of [Filter](#) objects

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Required: No

[maxResults](#)

The maximum number of agreements to return in the response.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

[nextToken](#)

A token to specify where to start pagination.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: `^[a-zA-Z0-9+/=]+$`

Required: No

sort

An object that contains the `SortBy` and `SortOrder` attributes.

Type: [Sort](#) object

Required: No

Response Syntax

```
{
  "agreementViewSummaries": [
    {
      "acceptanceTime": number,
      "acceptor": {
        "accountId": "string"
      },
      "agreementId": "string",
      "agreementType": "string",
      "endTime": number,
      "proposalSummary": {
        "offerId": "string",
        "resources": [
          {
            "id": "string",
            "type": "string"
          }
        ]
      },
      "proposer": {
        "accountId": "string"
      },
      "startTime": number,
      "status": "string"
    }
  ],
  "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

agreementViewSummaries

A summary of the agreement, including top-level attributes (for example, the agreement ID, version, proposer, and acceptor).

Type: Array of [AgreementViewSummary](#) objects

nextToken

The token used for pagination. The field is null if there are no more results.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: `^[a-zA-Z0-9+/=]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerErrorException

Unexpected error during processing of request.

HTTP Status Code: 500

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 400

ValidationException

The input fails to satisfy the constraints specified by the service.

HTTP Status Code: 400

Examples

Sample request

This example illustrates one usage of SearchAgreements.

```
{
  "catalog": "AWSMarketplace",
  "filters": [
    {
      "name": "PartyType",
      "values": ["Proposer"]
    },
    {
      "name": "AfterEndTime",
      "values": ["2019-10-08T00:00:00.000Z"]
    },
    {
      "name": "AcceptorAccountId",
      "values": ["123456789010"]
    }
  ]
}
```

Sample response

This example illustrates one usage of SearchAgreements.

```
{
  "agreementViewSummaries": [
    {
      "agreementId": "fEXAMPLE-0aa6-4e42-8715-6a1EXAMPLE95",
      "acceptor": {
        "accountId": "123456789010"
      },
      "proposer": {
        "accountId": "123456789010"
      },
      "startTime": "2019-10-08T21:40:43.644Z",
      "endTime": "2023-10-08T21:40:43.644Z",
      "acceptanceTime": "2019-10-08T00:00:00.000Z",
      "agreementType": "PurchaseAgreement",
      "proposalSummary": {
```

```
        "resources": [
            {
                "id": "0EXAMPLE-8ce8-4814-bcf1-636EXAMPLEb5",
                "type": "AmiProduct"
            }
        ],
        "offerId": "ABCDEFGHIJKLMN123"
    },
    "status": "ACTIVE"
}
],
"nextToken": null
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Metering API

The following actions are supported by AWS Marketplace Metering API:

- [BatchMeterUsage](#)
- [MeterUsage](#)
- [RegisterUsage](#)
- [ResolveCustomer](#)

BatchMeterUsage

Service: AWSMarketplace Metering

To post metering records for customers, SaaS applications call `BatchMeterUsage`, which is used for metering SaaS flexible consumption pricing (FCP). Identical requests are idempotent and can be retried with the same records or a subset of records. Each `BatchMeterUsage` request is for only one product. If you want to meter usage for multiple products, you must make multiple `BatchMeterUsage` calls.

Usage records should be submitted in quick succession following a recorded event. Usage records aren't accepted 6 hours or more after an event.

`BatchMeterUsage` can process up to 25 `UsageRecords` at a time, and each request must be less than 1 MB in size. Optionally, you can have multiple usage allocations for usage data that's split into buckets according to predefined tags.

`BatchMeterUsage` returns a list of `UsageRecordResult` objects, which have each `UsageRecord`. It also returns a list of `UnprocessedRecords`, which indicate errors on the service side that should be retried.

For AWS Regions that support `BatchMeterUsage`, see [BatchMeterUsage Region support](#).

Note

For an example of `BatchMeterUsage`, see [BatchMeterUsage code example](#) in the *AWS Marketplace Seller Guide*.

Request Syntax

```
{
  "ProductCode": "string",
  "UsageRecords": [
    {
      "CustomerIdentifier": "string",
      "Dimension": "string",
      "Quantity": number,
      "Timestamp": number,
      "UsageAllocations": [
        {
```

```
    "AllocatedUsageQuantity": number,
    "Tags": [
      {
        "Key": "string",
        "Value": "string"
      }
    ]
  }
]
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

ProductCode

Product code is used to uniquely identify a product in AWS Marketplace. The product code should be the same as the one used during the publishing of a new product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[-a-zA-Z0-9/=:_.*]*$`

Required: Yes

UsageRecords

The set of UsageRecords to submit. BatchMeterUsage accepts up to 25 UsageRecords at a time.

Type: Array of [UsageRecord](#) objects

Array Members: Minimum number of 0 items. Maximum number of 25 items.

Required: Yes

Response Syntax

```
{
  "Results": [
    {
      "MeteringRecordId": "string",
      "Status": "string",
      "UsageRecord": {
        "CustomerIdentifier": "string",
        "Dimension": "string",
        "Quantity": number,
        "Timestamp": number,
        "UsageAllocations": [
          {
            "AllocatedUsageQuantity": number,
            "Tags": [
              {
                "Key": "string",
                "Value": "string"
              }
            ]
          }
        ]
      }
    ]
  ],
  "UnprocessedRecords": [
    {
      "CustomerIdentifier": "string",
      "Dimension": "string",
      "Quantity": number,
      "Timestamp": number,
      "UsageAllocations": [
        {
          "AllocatedUsageQuantity": number,
          "Tags": [
            {
              "Key": "string",
              "Value": "string"
            }
          ]
        }
      ]
    }
  ]
}
```

```
    }  
  ]  
}  
]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Results

Contains all UsageRecords processed by BatchMeterUsage. These records were either honored by AWS Marketplace Metering Service or were invalid. Invalid records should be fixed before being resubmitted.

Type: Array of [UsageRecordResult](#) objects

UnprocessedRecords

Contains all UsageRecords that were not processed by BatchMeterUsage. This is a list of UsageRecords. You can retry the failed request by making another BatchMeterUsage call with this list as input in the BatchMeterUsageRequest.

Type: Array of [UsageRecord](#) objects

Array Members: Minimum number of 0 items. Maximum number of 25 items.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

DisabledApiException

The API is disabled in the Region.

HTTP Status Code: 400

InternalServiceErrorException

An internal error has occurred. Retry your request. If the problem persists, post a message with details on the AWS forums.

HTTP Status Code: 500

InvalidCustomerIdentifierException

You have metered usage for a `CustomerIdentifier` that does not exist.

HTTP Status Code: 400

InvalidProductCodeException

The product code passed does not match the product code used for publishing the product.

HTTP Status Code: 400

InvalidTagException

The tag is invalid, or the number of tags is greater than 5.

HTTP Status Code: 400

InvalidUsageAllocationsException

Sum of allocated usage quantities is not equal to the usage quantity.

HTTP Status Code: 400

InvalidUsageDimensionException

The usage dimension does not match one of the `UsageDimensions` associated with products.

HTTP Status Code: 400

ThrottlingException

The calls to the API are throttled.

HTTP Status Code: 400

TimestampOutOfBoundsException

The `timestamp` value passed in the `UsageRecord` is out of allowed range.

For `BatchMeterUsage`, if any of the records are outside of the allowed range, the entire batch is not processed. You must remove invalid records and try again.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

MeterUsage

Service: AWSMarketplace Metering

API to emit metering records. For identical requests, the API is idempotent and returns the metering record ID. This is used for metering flexible consumption pricing (FCP) Amazon Machine Images (AMI) and container products.

MeterUsage is authenticated on the buyer's AWS account using credentials from the Amazon EC2 instance, Amazon ECS task, or Amazon EKS pod.

MeterUsage can optionally include multiple usage allocations, to provide customers with usage data split into buckets by tags that you define (or allow the customer to define).

Usage records are expected to be submitted as quickly as possible after the event that is being recorded, and are not accepted more than 6 hours after the event.

For AWS Regions that support MeterUsage, see [MeterUsage Region support for Amazon EC2](#) and [MeterUsage Region support for Amazon ECS and Amazon EKS](#).

Request Syntax

```
{
  "DryRun": boolean,
  "ProductCode": "string",
  "Timestamp": number,
  "UsageAllocations": [
    {
      "AllocatedUsageQuantity": number,
      "Tags": [
        {
          "Key": "string",
          "Value": "string"
        }
      ]
    }
  ],
  "UsageDimension": "string",
  "UsageQuantity": number
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

ProductCode

Product code is used to uniquely identify a product in AWS Marketplace. The product code should be the same as the one used during the publishing of a new product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[-a-zA-Z0-9/=:_@]*$`

Required: Yes

Timestamp

Timestamp, in UTC, for which the usage is being reported. Your application can meter usage for up to six hours in the past. Make sure the timestamp value is not before the start of the software usage.

Type: Timestamp

Required: Yes

UsageDimension

It will be one of the fcp dimension name provided during the publishing of the product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[\s\S]+`

Required: Yes

DryRun

Checks whether you have the permissions required for the action, but does not make the request. If you have the permissions, the request returns `DryRunOperation`; otherwise, it returns `UnauthorizedException`. Defaults to `false` if not specified.

Type: Boolean

Required: No

UsageAllocations

The set of `UsageAllocations` to submit.

The sum of all `UsageAllocation` quantities must equal the `UsageQuantity` of the `MeterUsage` request, and each `UsageAllocation` must have a unique set of tags (include no tags).

Type: Array of [UsageAllocation](#) objects

Array Members: Minimum number of 1 item. Maximum number of 2500 items.

Required: No

UsageQuantity

Consumption value for the hour. Defaults to 0 if not specified.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

Required: No

Response Syntax

```
{
  "MeteringRecordId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

MeteringRecordId

Metering record id.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

CustomerNotEntitledException

Exception thrown when the customer does not have a valid subscription for the product.

HTTP Status Code: 400

DuplicateRequestException

A metering record has already been emitted by the same EC2 instance, ECS task, or EKS pod for the given {usageDimension, timestamp} with a different usageQuantity.

HTTP Status Code: 400

InternalServiceErrorException

An internal error has occurred. Retry your request. If the problem persists, post a message with details on the AWS forums.

HTTP Status Code: 500

InvalidEndpointRegionException

The endpoint being called is in a AWS Region different from your EC2 instance, ECS task, or EKS pod. The Region of the Metering Service endpoint and the AWS Region of the resource must match.

HTTP Status Code: 400

InvalidProductCodeException

The product code passed does not match the product code used for publishing the product.

HTTP Status Code: 400

InvalidTagException

The tag is invalid, or the number of tags is greater than 5.

HTTP Status Code: 400

InvalidUsageAllocationsException

Sum of allocated usage quantities is not equal to the usage quantity.

HTTP Status Code: 400

InvalidUsageDimensionException

The usage dimension does not match one of the UsageDimensions associated with products.

HTTP Status Code: 400

ThrottlingException

The calls to the API are throttled.

HTTP Status Code: 400

TimestampOutOfBoundsException

The timestamp value passed in the UsageRecord is out of allowed range.

For BatchMeterUsage, if any of the records are outside of the allowed range, the entire batch is not processed. You must remove invalid records and try again.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

RegisterUsage

Service: AWSMarketplace Metering

Paid container software products sold through AWS Marketplace must integrate with the AWS Marketplace Metering Service and call the `RegisterUsage` operation for software entitlement and metering. Free and BYOL products for Amazon ECS or Amazon EKS aren't required to call `RegisterUsage`, but you may choose to do so if you would like to receive usage data in your seller reports. The sections below explain the behavior of `RegisterUsage`. `RegisterUsage` performs two primary functions: metering and entitlement.

- *Entitlement*: `RegisterUsage` allows you to verify that the customer running your paid software is subscribed to your product on AWS Marketplace, enabling you to guard against unauthorized use. Your container image that integrates with `RegisterUsage` is only required to guard against unauthorized use at container startup, as such a `CustomerNotSubscribedException` or `PlatformNotSupportedException` will only be thrown on the initial call to `RegisterUsage`. Subsequent calls from the same Amazon ECS task instance (e.g. task-id) or Amazon EKS pod will not throw a `CustomerNotSubscribedException`, even if the customer unsubscribes while the Amazon ECS task or Amazon EKS pod is still running.
- *Metering*: `RegisterUsage` meters software use per ECS task, per hour, or per pod for Amazon EKS with usage prorated to the second. A minimum of 1 minute of usage applies to tasks that are short lived. For example, if a customer has a 10 node Amazon ECS or Amazon EKS cluster and a service configured as a Daemon Set, then Amazon ECS or Amazon EKS will launch a task on all 10 cluster nodes and the customer will be charged for 10 tasks. Software metering is handled by the AWS Marketplace metering control plane—your software is not required to perform metering-specific actions other than to call `RegisterUsage` to commence metering. The AWS Marketplace metering control plane will also bill customers for running ECS tasks and Amazon EKS pods, regardless of the customer's subscription state, which removes the need for your software to run entitlement checks at runtime. For containers, `RegisterUsage` should be called immediately at launch. If you don't register the container within the first 6 hours of the launch, AWS Marketplace Metering Service doesn't provide any metering guarantees for previous months. Metering will continue, however, for the current month forward until the container ends. `RegisterUsage` is for metering paid hourly container products.

For AWS Regions that support `RegisterUsage`, see [RegisterUsage Region support](#).

Request Syntax

```
{  
  "Nonce": "string",  
  "ProductCode": "string",  
  "PublicKeyVersion": number  
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

ProductCode

Product code is used to uniquely identify a product in AWS Marketplace. The product code should be the same as the one used during the publishing of a new product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[-a-zA-Z0-9/=:_.*]*$`

Required: Yes

PublicKeyVersion

Public Key Version provided by AWS Marketplace

Type: Integer

Valid Range: Minimum value of 1.

Required: Yes

Nonce

(Optional) To scope down the registration to a specific running software instance and guard against replay attacks.

Type: String

Length Constraints: Maximum length of 255.

Pattern: `[\s\S]*`

Required: No

Response Syntax

```
{
  "PublicKeyRotationTimestamp": number,
  "Signature": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[PublicKeyRotationTimestamp](#)

(Optional) Only included when public key version has expired

Type: Timestamp

[Signature](#)

JWT Token

Type: String

Pattern: `[\s\S]+`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

CustomerNotEntitledException

Exception thrown when the customer does not have a valid subscription for the product.

HTTP Status Code: 400

DisabledApiException

The API is disabled in the Region.

HTTP Status Code: 400

InternalServiceErrorException

An internal error has occurred. Retry your request. If the problem persists, post a message with details on the AWS forums.

HTTP Status Code: 500

InvalidProductCodeException

The product code passed does not match the product code used for publishing the product.

HTTP Status Code: 400

InvalidPublicKeyVersionException

Public Key version is invalid.

HTTP Status Code: 400

InvalidRegionException

`RegisterUsage` must be called in the same AWS Region the ECS task was launched in. This prevents a container from hardcoding a Region (e.g. `withRegion("us-east-1")`) when calling `RegisterUsage`.

HTTP Status Code: 400

PlatformNotSupportedException

AWS Marketplace does not support metering usage from the underlying platform. Currently, Amazon ECS, Amazon EKS, and AWS Fargate are supported.

HTTP Status Code: 400

ThrottlingException

The calls to the API are throttled.

HTTP Status Code: 400

Examples

Example

Below are the sample request and response for RegisterUsage

Sample Request

```
{
  "ProductCode" : "cqcvf9f0ugw8rkbgmf1c9dxyz",
  "PublicKeyVersion": 1,
  "Nonce": "2ead20e4-3e6d-42cd-8f56-24f02d1cc4e1"
}
```

Sample Response

```
{
  "PublicKeyRotationTimestamp": null,
  "Signature": "eyJhbGciOiJIUzI1Ni..."
}

// Where the signature is composed of 3 dot-separated,
// base-64 URL Encoded sections.
// e.g. eyJhbGciOiJIUzI1Ni...

// Section 1: Header/Algorithm

{
  "alg": "PS256",
  "typ": "JWT"
}

// Section 2: Payload

{
  "ProductCode" : "cqcvf9f0ugw8rkbgmf1c9dxyz",
  "PublicKeyVersion": 1,
  "Nonce": "2ead20e4-3e6d-42cd-8f56-24f02d1cc4e1",
  "PublicKeyRotationTimestamp": null
}
```

```
}  
  
// Section 3: RSA-PSS SHA256 signature  
  
"rr09Q4FEi3gweH3X4lrt2okf5zwIatUUwERlw016wTy_21Nv8S..."
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ResolveCustomer

Service: AWSMarketplace Metering

ResolveCustomer is called by a SaaS application during the registration process. When a buyer visits your website during the registration process, the buyer submits a registration token through their browser. The registration token is resolved through this API to obtain a CustomerIdentifier along with the CustomerAWSAccountId and ProductCode.

Note

To successfully resolve the token, the API must be called from the account that was used to publish the SaaS application. For an example of using ResolveCustomer, see [ResolveCustomer code example](#) in the *AWS Marketplace Seller Guide*.

Permission is required for this operation. Your IAM role or user performing this operation requires a policy to allow the `aws-marketplace:ResolveCustomer` action. For more information, see [Actions, resources, and condition keys for AWS Marketplace Metering Service](#) in the *Service Authorization Reference*.

For AWS Regions that support ResolveCustomer, see [ResolveCustomer Region support](#).

Request Syntax

```
{
  "RegistrationToken": "string"
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

RegistrationToken

When a buyer visits your website during the registration process, the buyer submits a registration token through the browser. The registration token is resolved to obtain a `CustomerIdentifier` along with the `CustomerAWSAccountId` and `ProductCode`.

Type: String

Pattern: `[\s\S]+`

Required: Yes

Response Syntax

```
{
  "CustomerAWSAccountId": "string",
  "CustomerIdentifier": "string",
  "ProductCode": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CustomerAWSAccountId

The `CustomerAWSAccountId` provides the AWS account ID associated with the `CustomerIdentifier` for the individual customer.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[0-9]+$`

CustomerIdentifier

The `CustomerIdentifier` is used to identify an individual customer in your application. Calls to `BatchMeterUsage` require `CustomerIdentifiers` for each `UsageRecord`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[\s\S]+`

ProductCode

The product code is returned to confirm that the buyer is registering for your product. Subsequent `BatchMeterUsage` calls should be made using this product code.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[-a-zA-Z0-9/=:_.*]*$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

DisabledApiException

The API is disabled in the Region.

HTTP Status Code: 400

ExpiredTokenException

The submitted registration token has expired. This can happen if the buyer's browser takes too long to redirect to your page, the buyer has resubmitted the registration token, or your application has held on to the registration token for too long. Your SaaS registration website should redeem this token as soon as it is submitted by the buyer's browser.

HTTP Status Code: 400

InternalServiceErrorException

An internal error has occurred. Retry your request. If the problem persists, post a message with details on the AWS forums.

HTTP Status Code: 500

InvalidTokenException

Registration token is invalid.

HTTP Status Code: 400

ThrottlingException

The calls to the API are throttled.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Entitlement API

The following actions are supported by AWS Marketplace Entitlement API:

- [GetEntitlements](#)

GetEntitlements

Service: AWS Marketplace Entitlement Service

GetEntitlements retrieves entitlement values for a given product. The results can be filtered based on customer identifier or product dimensions.

Request Syntax

```
{
  "Filter": {
    "string" : [ "string" ]
  },
  "MaxResults": number,
  "NextToken": "string",
  "ProductCode": "string"
}
```

Request Parameters

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

ProductCode

Product code is used to uniquely identify a product in AWS Marketplace. The product code will be provided by AWS Marketplace when the product listing is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: Yes

Filter

Filter is used to return entitlements for a specific customer or for a specific dimension. Filters are described as keys mapped to a lists of values. Filtered requests are *unioned* for each value in the value list, and then *intersected* for each filter key.

Type: String to array of strings map

Valid Keys: CUSTOMER_IDENTIFIER | DIMENSION

Array Members: Minimum number of 1 item.

Required: No

MaxResults

The maximum number of items to retrieve from the GetEntitlements operation. For pagination, use the NextToken field in subsequent calls to GetEntitlements.

Type: Integer

Required: No

NextToken

For paginated calls to GetEntitlements, pass the NextToken from the previous GetEntitlementsResult.

Type: String

Pattern: \S+

Required: No

Response Syntax

```
{
  "Entitlements": [
    {
      "CustomerIdentifier": "string",
      "Dimension": "string",
      "ExpirationDate": number,
      "ProductCode": "string",
      "Value": {
        "BooleanValue": boolean,
        "DoubleValue": number,
        "IntegerValue": number,
        "StringValue": "string"
      }
    }
  ]
}
```

```
    }  
  ],  
  "NextToken": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Entitlements

The set of entitlements found through the GetEntitlements operation. If the result contains an empty set of entitlements, NextToken might still be present and should be used.

Type: Array of [Entitlement](#) objects

Array Members: Minimum number of 0 items.

NextToken

For paginated results, use NextToken in subsequent calls to GetEntitlements. If the result contains an empty set of entitlements, NextToken might still be present and should be used.

Type: String

Pattern: \S+

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

InternalServerErrorException

An internal error has occurred. Retry your request. If the problem persists, post a message with details on the AWS forums.

HTTP Status Code: 500

InvalidParameterException

One or more parameters in your request was invalid.

HTTP Status Code: 400

ThrottlingException

The calls to the GetEntitlements API are throttled.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Deployment API

The following actions are supported by AWS Marketplace Deployment API:

- [ListTagsForResource](#)
- [PutDeploymentParameter](#)
- [TagResource](#)
- [UntagResource](#)

ListTagsForResource

Service: AWS Marketplace Deployment Service

Lists all tags that have been added to a deployment parameter resource.

Request Syntax

```
GET /tags/resourceArn HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

resourceArn

The Amazon Resource Name (ARN) associated with the deployment parameter resource you want to list tags on.

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "tags": {
    "string" : "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

tags

A map of key-value pairs, where each pair represents a tag present on the resource.

Type: String to string map

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

InternalServerErrorException

There was an internal service exception.

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

PutDeploymentParameter

Service: AWS Marketplace Deployment Service

Creates or updates a deployment parameter and is targeted by `catalog` and `agreementId`.

Request Syntax

```
POST /catalogs/catalog/products/productId/deployment-parameters HTTP/1.1
Content-type: application/json
```

```
{
  "agreementId": "string",
  "clientToken": "string",
  "deploymentParameter": {
    "name": "string",
    "secretString": "string"
  },
  "expirationDate": "string",
  "tags": {
    "string" : "string"
  }
}
```

URI Request Parameters

The request uses the following URI parameters.

catalog

The catalog related to the request. Fixed value: `AWSMarketplace`

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z_-]+$`

Required: Yes

productId

The product for which AWS Marketplace will save secrets for the buyer's account.

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: Yes

Request Body

The request accepts the following data in JSON format.

agreementId

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: Yes

deploymentParameter

The deployment parameter targeted to the acceptor of an agreement for which to create the AWS Secret Manager resource.

Type: [DeploymentParameterInput](#) object

Required: Yes

clientToken

The idempotency token for deployment parameters. A unique identifier for the new version.

Note

This field is not required if you're calling using an AWS SDK. Otherwise, a `clientToken` must be provided with the request.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^[a-zA-Z0-9/_+=.:@-]+$`

Required: No

expirationDate

The date when deployment parameters expire and are scheduled for deletion.

Type: Timestamp

Required: No

tags

A map of key-value pairs, where each pair represents a tag saved to the resource. Tags will only be applied for create operations, and they'll be ignored if the resource already exists.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^[a-zA-Z0-9/_+=.:@-]+$`

Value Length Constraints: Minimum length of 1. Maximum length of 256.

Value Pattern: `^[a-zA-Z0-9/_+=.:@-]+$`

Required: No

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "agreementId": "string",
  "deploymentParameterId": "string",
  "resourceArn": "string",
  "tags": {
    "string" : "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

agreementId

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

deploymentParameterId

The unique identifier of the deployment parameter.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `^dp-[a-zA-Z0-9]+$`

resourceArn

The Amazon Resource Name (ARN) associated with the deployment parameter resource you want to create or update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

tags

A map of key-value pairs, where each pair represents a tag saved to the resource. Tags will only be applied for create operations, and they'll be ignored if the resource already exists.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^[a-zA-Z0-9/_+=.:@-]+$`

Value Length Constraints: Minimum length of 1. Maximum length of 256.

Value Pattern: `^[a-zA-Z0-9/_+=.:@-]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

ConflictException

The request configuration has conflicts. For details, see the accompanying error message.

HTTP Status Code: 409

InternalServerError

There was an internal service exception.

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP Status Code: 404

ServiceQuotaExceededException

The maximum number of requests per account has been exceeded.

HTTP Status Code: 402

ThrottlingException

Too many requests.

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

TagResource

Service: AWS Marketplace Deployment Service

Tags a resource.

Request Syntax

```
POST /tags/resourceArn HTTP/1.1  
Content-type: application/json
```

```
{  
  "tags": {  
    "string" : "string"  
  }  
}
```

URI Request Parameters

The request uses the following URI parameters.

resourceArn

The Amazon Resource Name (ARN) associated with the resource you want to tag.

Required: Yes

Request Body

The request accepts the following data in JSON format.

tags

A map of key-value pairs, where each pair represents a tag present on the resource.

Type: String to string map

Required: No

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

ConflictException

The request configuration has conflicts. For details, see the accompanying error message.

HTTP Status Code: 409

InternalServerErrorException

There was an internal service exception.

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UntagResource

Service: AWS Marketplace Deployment Service

Removes a tag or list of tags from a resource.

Request Syntax

```
DELETE /tags/resourceArn?tagKeys=tagKeys HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

resourceArn

The Amazon Resource Name (ARN) associated with the resource you want to remove the tag from.

Required: Yes

tagKeys

A list of key names of tags to be removed.

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

ConflictException

The request configuration has conflicts. For details, see the accompanying error message.

HTTP Status Code: 409

InternalServerErrorException

There was an internal service exception.

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Reporting API

The following actions are supported by the AWS Marketplace Reporting API:

- [GetBuyerDashboard](#)

GetBuyerDashboard

Service: AWS Marketplace Reporting Service

Generates an embedding URL for an Amazon QuickSight dashboard for an anonymous user.

Note

This API is available only to AWS Organization management accounts or delegated administrators registered for the procurement insights (procurement-insights.marketplace.amazonaws.com) feature.

The following rules apply to a generated URL:

- It contains a temporary bearer token, valid for 5 minutes after it is generated. Once redeemed within that period, it cannot be re-used again.
- It has a session lifetime of one hour. The 5-minute validity period runs separately from the session lifetime.

Request Syntax

```
POST /getBuyerDashboard HTTP/1.1
Content-type: application/json

{
  "dashboardIdentifier": "string",
  "embeddingDomains": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

dashboardIdentifier

The ARN of the requested dashboard.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1023.

Pattern: `arn:aws:aws-marketplace::[0-9]{12}:AWSMarketplace/ReportingData/(Agreement_V1/Dashboard/AgreementSummary_V1|BillingEvent_V1/Dashboard/CostAnalysis_V1)`

Required: Yes

embeddingDomains

Fully qualified domains that you add to the allow list for access to the generated URL that is then embedded. You can list up to two domains or subdomains in each API call. To include all subdomains under a specific domain, use *. For example, `https://*.amazon.com` includes all subdomains under `https://aws.amazon.com`.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 2 items.

Length Constraints: Minimum length of 1. Maximum length of 2000.

Pattern: `(https://[a-zA-Z\.*0-9\-_]+[\.\.]{1}[a-zA-Z]{1,}[a-zA-Z0-9&?/_\-=]*[a-zA-Z*0-9/]+|http[s]*://localhost(:[0-9]{1,5})?)`

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "dashboardIdentifier": "string",
  "embeddingDomains": [ "string" ],
  "embedUrl": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

dashboardIdentifier

The ARN of the returned dashboard.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1023.

Pattern: `arn:aws:aws-marketplace::[0-9]{12}:AWSMarketplace/ReportingData/(Agreement_V1/Dashboard/AgreementSummary_V1|BillingEvent_V1/Dashboard/CostAnalysis_V1)`

embeddingDomains

The fully qualified domains specified in the request. The domains enable access to the generated URL that is then embedded. You can list up to two domains or subdomains in each API call. To include all subdomains under a specific domain, use *. For example, `https://*.amazon.com` includes all subdomains under `https://aws.amazon.com`.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 2 items.

Length Constraints: Minimum length of 1. Maximum length of 2000.

Pattern: `(https://[a-zA-Z\.*0-9\-_]+[\.\.]{1}[a-zA-Z]{1,}[a-zA-Z0-9&?/-_=]*[a-zA-Z*0-9/]+|http[s]*://localhost(:[0-9]{1,5})?)`

embedUrl

The dashboard's embedding URL.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

BadRequestException

The request is malformed, or it contains an error such as an invalid parameter. Ensure the request has all required parameters.

HTTP Status Code: 400

InternalServerErrorException

The operation failed due to a server error.

HTTP Status Code: 500

UnauthorizedException

You do not have permission to perform this action.

HTTP Status Code: 401

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Data Types

The following data types are supported by AWS Marketplace Catalog Service:

- [AmiProductEntityIdFilter](#)
- [AmiProductFilters](#)
- [AmiProductLastModifiedDateFilter](#)
- [AmiProductLastModifiedDateFilterDateRange](#)
- [AmiProductSort](#)
- [AmiProductSummary](#)
- [AmiProductTitleFilter](#)
- [AmiProductVisibilityFilter](#)
- [BatchDescribeErrorDetail](#)
- [Change](#)
- [ChangeSetSummaryListItem](#)
- [ChangeSummary](#)
- [ContainerProductEntityIdFilter](#)
- [ContainerProductFilters](#)
- [ContainerProductLastModifiedDateFilter](#)
- [ContainerProductLastModifiedDateFilterDateRange](#)
- [ContainerProductSort](#)
- [ContainerProductSummary](#)
- [ContainerProductTitleFilter](#)
- [ContainerProductVisibilityFilter](#)
- [DataProductEntityIdFilter](#)
- [DataProductFilters](#)
- [DataProductLastModifiedDateFilter](#)
- [DataProductLastModifiedDateFilterDateRange](#)
- [DataProductSort](#)
- [DataProductSummary](#)
- [DataProductTitleFilter](#)

- [DataProductVisibilityFilter](#)
- [Entity](#)
- [EntityDetail](#)
- [EntityRequest](#)
- [EntitySummary](#)
- [EntityTypeFilters](#)
- [EntityTypeSort](#)
- [ErrorDetail](#)
- [Filter](#)
- [OfferAvailabilityEndDateFilter](#)
- [OfferAvailabilityEndDateFilterDateRange](#)
- [OfferBuyerAccountsFilter](#)
- [OfferEntityIdFilter](#)
- [OfferFilters](#)
- [OfferLastModifiedDateFilter](#)
- [OfferLastModifiedDateFilterDateRange](#)
- [OfferNameFilter](#)
- [OfferProductIdFilter](#)
- [OfferReleaseDateFilter](#)
- [OfferReleaseDateFilterDateRange](#)
- [OfferResaleAuthorizationIdFilter](#)
- [OfferSort](#)
- [OfferStateFilter](#)
- [OfferSummary](#)
- [OfferTargetingFilter](#)
- [ResaleAuthorizationAvailabilityEndDateFilter](#)
- [ResaleAuthorizationAvailabilityEndDateFilterDateRange](#)
- [ResaleAuthorizationCreatedDateFilter](#)
- [ResaleAuthorizationCreatedDateFilterDateRange](#)
- [ResaleAuthorizationEntityIdFilter](#)

- [ResaleAuthorizationFilters](#)
- [ResaleAuthorizationLastModifiedDateFilter](#)
- [ResaleAuthorizationLastModifiedDateFilterDateRange](#)
- [ResaleAuthorizationManufacturerAccountIdFilter](#)
- [ResaleAuthorizationManufacturerLegalNameFilter](#)
- [ResaleAuthorizationNameFilter](#)
- [ResaleAuthorizationOfferExtendedStatusFilter](#)
- [ResaleAuthorizationProductIdFilter](#)
- [ResaleAuthorizationProductNameFilter](#)
- [ResaleAuthorizationResellerAccountIDFilter](#)
- [ResaleAuthorizationResellerLegalNameFilter](#)
- [ResaleAuthorizationSort](#)
- [ResaleAuthorizationStatusFilter](#)
- [ResaleAuthorizationSummary](#)
- [SaaSProductEntityIdFilter](#)
- [SaaSProductFilters](#)
- [SaaSProductLastModifiedDateFilter](#)
- [SaaSProductLastModifiedDateFilterDateRange](#)
- [SaaSProductSort](#)
- [SaaSProductSummary](#)
- [SaaSProductTitleFilter](#)
- [SaaSProductVisibilityFilter](#)
- [Sort](#)
- [Tag](#)

The following data types are supported by AWS Marketplace Agreement Service:

- [AcceptedTerm](#)
- [Acceptor](#)
- [AgreementViewSummary](#)
- [ByolPricingTerm](#)

- [ConfigurableUpfrontPricingTerm](#)
- [ConfigurableUpfrontPricingTermConfiguration](#)
- [ConfigurableUpfrontRateCardItem](#)
- [Constraints](#)
- [Dimension](#)
- [DocumentItem](#)
- [EstimatedCharges](#)
- [Filter](#)
- [FixedUpfrontPricingTerm](#)
- [FreeTrialPricingTerm](#)
- [GrantItem](#)
- [LegalTerm](#)
- [PaymentScheduleTerm](#)
- [ProposalSummary](#)
- [Proposer](#)
- [RateCardItem](#)
- [RecurringPaymentTerm](#)
- [RenewalTerm](#)
- [RenewalTermConfiguration](#)
- [Resource](#)
- [ScheduleItem](#)
- [Selector](#)
- [Sort](#)
- [SupportTerm](#)
- [UsageBasedPricingTerm](#)
- [UsageBasedRateCardItem](#)
- [ValidationExceptionField](#)
- [ValidityTerm](#)

The following data types are supported by AWS Marketplace Metering Service:

- [Tag](#)
- [UsageAllocation](#)
- [UsageRecord](#)
- [UsageRecordResult](#)

The following data types are supported by AWS Marketplace Entitlement Service:

- [Entitlement](#)
- [EntitlementValue](#)

The following data types are supported by AWS Marketplace Deployment Service:

- [DeploymentParameterInput](#)

The following data types are supported by AWS Marketplace Reporting Service:

AWS Marketplace Catalog API

The following data types are supported by AWS Marketplace Catalog API:

- [AmiProductEntityIdFilter](#)
- [AmiProductFilters](#)
- [AmiProductLastModifiedDateFilter](#)
- [AmiProductLastModifiedDateFilterDateRange](#)
- [AmiProductSort](#)
- [AmiProductSummary](#)
- [AmiProductTitleFilter](#)
- [AmiProductVisibilityFilter](#)
- [BatchDescribeErrorDetail](#)
- [Change](#)
- [ChangeSetSummaryListItem](#)
- [ChangeSummary](#)
- [ContainerProductEntityIdFilter](#)

- [ContainerProductFilters](#)
- [ContainerProductLastModifiedDateFilter](#)
- [ContainerProductLastModifiedDateFilterDateRange](#)
- [ContainerProductSort](#)
- [ContainerProductSummary](#)
- [ContainerProductTitleFilter](#)
- [ContainerProductVisibilityFilter](#)
- [DataProductEntityIdFilter](#)
- [DataProductFilters](#)
- [DataProductLastModifiedDateFilter](#)
- [DataProductLastModifiedDateFilterDateRange](#)
- [DataProductSort](#)
- [DataProductSummary](#)
- [DataProductTitleFilter](#)
- [DataProductVisibilityFilter](#)
- [Entity](#)
- [EntityDetail](#)
- [EntityRequest](#)
- [EntitySummary](#)
- [EntityTypeFilters](#)
- [EntityTypeSort](#)
- [ErrorDetail](#)
- [Filter](#)
- [OfferAvailabilityEndDateFilter](#)
- [OfferAvailabilityEndDateFilterDateRange](#)
- [OfferBuyerAccountsFilter](#)
- [OfferEntityIdFilter](#)
- [OfferFilters](#)
- [OfferLastModifiedDateFilter](#)
- [OfferLastModifiedDateFilterDateRange](#)

- [OfferNameFilter](#)
- [OfferProductIdFilter](#)
- [OfferReleaseDateFilter](#)
- [OfferReleaseDateFilterDateRange](#)
- [OfferResaleAuthorizationIdFilter](#)
- [OfferSort](#)
- [OfferStateFilter](#)
- [OfferSummary](#)
- [OfferTargetingFilter](#)
- [ResaleAuthorizationAvailabilityEndDateFilter](#)
- [ResaleAuthorizationAvailabilityEndDateFilterDateRange](#)
- [ResaleAuthorizationCreatedDateFilter](#)
- [ResaleAuthorizationCreatedDateFilterDateRange](#)
- [ResaleAuthorizationEntityIdFilter](#)
- [ResaleAuthorizationFilters](#)
- [ResaleAuthorizationLastModifiedDateFilter](#)
- [ResaleAuthorizationLastModifiedDateFilterDateRange](#)
- [ResaleAuthorizationManufacturerAccountIdFilter](#)
- [ResaleAuthorizationManufacturerLegalNameFilter](#)
- [ResaleAuthorizationNameFilter](#)
- [ResaleAuthorizationOfferExtendedStatusFilter](#)
- [ResaleAuthorizationProductIdFilter](#)
- [ResaleAuthorizationProductNameFilter](#)
- [ResaleAuthorizationResellerAccountIdFilter](#)
- [ResaleAuthorizationResellerLegalNameFilter](#)
- [ResaleAuthorizationSort](#)
- [ResaleAuthorizationStatusFilter](#)
- [ResaleAuthorizationSummary](#)
- [SaaSProductEntityIdFilter](#)
- [SaaSProductFilters](#)

- [SaaSProductLastModifiedDateFilter](#)
- [SaaSProductLastModifiedDateFilterDateRange](#)
- [SaaSProductSort](#)
- [SaaSProductSummary](#)
- [SaaSProductTitleFilter](#)
- [SaaSProductVisibilityFilter](#)
- [Sort](#)
- [Tag](#)

AmiProductEntityIdFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on entity id of an AMI product.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique entity id values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for AMI products. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

EntityId

Unique identifier for the AMI product.

Type: [AmiProductEntityIdFilter](#) object

Required: No

LastModifiedDate

The last date on which the AMI product was modified.

Type: [AmiProductLastModifiedDateFilter](#) object

Required: No

ProductTitle

The title of the AMI product.

Type: [AmiProductTitleFilter](#) object

Required: No

Visibility

The visibility of the AMI product.

Type: [AmiProductVisibilityFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering based on the last modified date of AMI products.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Dates between which the AMI product was last modified.

Type: [AmiProductLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Object that contains date range of the last modified date to be filtered on. You can optionally provide a `BeforeValue` and/or `AfterValue`. Both are inclusive.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Date after which the AMI product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Date before which the AMI product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductSort

Service: AWS Marketplace Catalog Service

Objects that allows sorting on AMI products based on certain fields and sorting order.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Field to sort the AMI products by.

Type: String

Valid Values: EntityId | LastModifiedDate | ProductTitle | Visibility

Required: No

SortOrder

The sorting order. Can be ASCENDING or DESCENDING. The default value is DESCENDING.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductSummary

Service: AWS Marketplace Catalog Service

Object that contains summarized information about an AMI product.

Contents

Note

In the following list, the required parameters are described first.

ProductTitle

The title of the AMI product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Visibility

The lifecycle of the AMI product.

Type: String

Valid Values: `Limited` | `Public` | `Restricted` | `Draft`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

AmiProductTitleFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on product title.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique product title values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

A string that will be the `wildCard` input for product tile filter. It matches the provided value as a substring in the actual value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AmiProductVisibilityFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on the visibility of the product in the AWS Marketplace.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique visibility values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Valid Values: Limited | Public | Restricted | Draft

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

BatchDescribeErrorDetail

Service: AWS Marketplace Catalog Service

An object that contains an error code and error message.

Contents

Note

In the following list, the required parameters are described first.

ErrorCode

The error code returned.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 72.

Pattern: `^[a-zA-Z_]+$`

Required: No

ErrorMessage

The error message returned.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Change

Service: AWS Marketplace Catalog Service

An object that contains the `ChangeType`, `Details`, and `Entity`.

Contents

Note

In the following list, the required parameters are described first.

ChangeType

Change types are single string values that describe your intention for the change. Each change type is unique for each `EntityType` provided in the change's scope. For more information about change types available for single-AMI products, see [Working with single-AMI products](#). Also, for more information about change types available for container-based products, see [Working with container products](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[A-Z][\w]*$`

Required: Yes

Entity

The entity to be changed.

Type: [Entity](#) object

Required: Yes

ChangeName

Optional name for the change.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 72.

Pattern: `^[a-zA-Z]$`

Required: No

Details

This object contains details specific to the change type of the requested change. For more information about change types available for single-AMI products, see [Working with single-AMI products](#). Also, for more information about change types available for container-based products, see [Working with container products](#).

Type: String

Length Constraints: Minimum length of 2. Maximum length of 16384.

Pattern: `^[\\s]*\\{[\\s\\S]*\\}[\\s]*$`

Required: No

DetailsDocument

Alternative field that accepts a JSON value instead of a string for ChangeType details. You can use either `Details` or `DetailsDocument`, but not both.

To download the "DetailsDocument" shapes, see the [Python](#) and [Java](#) shapes on GitHub.

Type: JSON value

Required: No

EntityTags

The tags associated with the change.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ChangeSetSummaryListItem

Service: AWS Marketplace Catalog Service

A summary of a change set returned in a list of change sets when the `ListChangeSets` action is called.

Contents

Note

In the following list, the required parameters are described first.

ChangeSetArn

The ARN associated with the unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

Required: No

ChangeSetId

The unique identifier for a change set.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

Required: No

ChangeSetName

The non-unique name for the change set.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^[\\w\\s+=.:@-]+$`

Required: No

EndTime

The time, in ISO 8601 format (2018-02-27T13:45:22Z), when the change set was finished.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\\d]{4})\\-(1[0-2]|0[1-9])\\-(3[01]|0[1-9]|[12][\\d])T(2[0-3]|[01][\\d]):(0-5)[\\d]):(0-5)[\\d]Z$`

Required: No

EntityIdList

This object is a list of entity IDs (string) that are a part of a change set. The entity ID list is a maximum of 20 entities. It must contain at least one entity.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-]+$`

Required: No

FailureCode

Returned if the change set is in FAILED status. Can be either CLIENT_ERROR, which means that there are issues with the request (see the ErrorDetailList of DescribeChangeSet), or SERVER_FAULT, which means that there is a problem in the system, and you should retry your request.

Type: String

Valid Values: CLIENT_ERROR | SERVER_FAULT

Required: No

StartTime

The time, in ISO 8601 format (2018-02-27T13:45:22Z), when the change set was started.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|01[\d]):(0-5[\d]):(0-5[\d])Z$`

Required: No

Status

The current status of the change set.

Type: String

Valid Values: PREPARING | APPLYING | SUCCEEDED | CANCELLED | FAILED

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ChangeSummary

Service: AWS Marketplace Catalog Service

This object is a container for common summary information about the change. The summary doesn't contain the whole change structure.

Contents

Note

In the following list, the required parameters are described first.

ChangeName

Optional name for the change.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 72.

Pattern: `^[a-zA-Z]$`

Required: No

ChangeType

The type of the change.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[A-Z][\w]*$`

Required: No

Details

This object contains details specific to the change type of the requested change.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 16384.

Pattern: `^\s*\{[\s\S]*\}\s*$`

Required: No

DetailsDocument

The JSON value of the details specific to the change type of the requested change.

To download the "DetailsDocument" shapes, see the [Python](#) and [Java](#) shapes on GitHub.

Type: JSON value

Required: No

Entity

The entity to be changed.

Type: [Entity](#) object

Required: No

ErrorDetailList

An array of `ErrorDetail` objects associated with the change.

Type: Array of [ErrorDetail](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductEntityIdFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on entity id of a container product.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique entity id values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for container products. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

EntityId

Unique identifier for the container product.

Type: [ContainerProductEntityIdFilter](#) object

Required: No

LastModifiedDate

The last date on which the container product was modified.

Type: [ContainerProductLastModifiedDateFilter](#) object

Required: No

ProductTitle

The title of the container product.

Type: [ContainerProductTitleFilter](#) object

Required: No

Visibility

The visibility of the container product.

Type: [ContainerProductVisibilityFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering based on the last modified date of container products.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Dates between which the container product was last modified.

Type: [ContainerProductLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Object that contains date range of the last modified date to be filtered on. You can optionally provide a `BeforeValue` and/or `AfterValue`. Both are inclusive.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Date after which the container product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Date before which the container product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductSort

Service: AWS Marketplace Catalog Service

Objects that allows sorting on container products based on certain fields and sorting order.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Field to sort the container products by.

Type: String

Valid Values: EntityId | LastModifiedDate | ProductTitle | Visibility

Required: No

SortOrder

The sorting order. Can be ASCENDING or DESCENDING. The default value is DESCENDING.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductSummary

Service: AWS Marketplace Catalog Service

Object that contains summarized information about a container product.

Contents

Note

In the following list, the required parameters are described first.

ProductTitle

The title of the container product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Visibility

The lifecycle of the product.

Type: String

Valid Values: `Limited` | `Public` | `Restricted` | `Draft`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

ContainerProductTitleFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on product title.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique product title values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

A string that will be the `wildCard` input for product tile filter. It matches the provided value as a substring in the actual value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContainerProductVisibilityFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on the visibility of the product in the AWS Marketplace.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique visibility values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Valid Values: Limited | Public | Restricted | Draft

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductEntityIdFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on entity id of a data product.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique entity id values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for data products. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

EntityId

Unique identifier for the data product.

Type: [DataProductEntityIdFilter](#) object

Required: No

LastModifiedDate

The last date on which the data product was modified.

Type: [DataProductLastModifiedDateFilter](#) object

Required: No

ProductTitle

The title of the data product.

Type: [DataProductTitleFilter](#) object

Required: No

Visibility

The visibility of the data product.

Type: [DataProductVisibilityFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering based on the last modified date of data products.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Dates between which the data product was last modified.

Type: [DataProductLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Object that contains date range of the last modified date to be filtered on. You can optionally provide a `BeforeValue` and/or `AfterValue`. Both are inclusive.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Date after which the data product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Date before which the data product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductSort

Service: AWS Marketplace Catalog Service

Objects that allows sorting on data products based on certain fields and sorting order.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Field to sort the data products by.

Type: String

Valid Values: EntityId | ProductTitle | Visibility | LastModifiedDate

Required: No

SortOrder

The sorting order. Can be ASCENDING or DESCENDING. The default value is DESCENDING.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductSummary

Service: AWS Marketplace Catalog Service

Object that contains summarized information about a data product.

Contents

Note

In the following list, the required parameters are described first.

ProductTitle

The title of the data product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Visibility

The lifecycle of the data product.

Type: String

Valid Values: `Limited` | `Public` | `Restricted` | `Unavailable` | `Draft`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

DataProductTitleFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on product title.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique product title values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

A string that will be the `wildCard` input for product tile filter. It matches the provided value as a substring in the actual value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DataProductVisibilityFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on the visibility of the product in the AWS Marketplace.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique visibility values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Valid Values: Limited | Public | Restricted | Unavailable | Draft

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Entity

Service: AWS Marketplace Catalog Service

An entity contains data that describes your product, its supported features, and how it can be used or launched by your customer.

Contents

Note

In the following list, the required parameters are described first.

Type

The type of entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: Yes

Identifier

The identifier for the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-@]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntityDetail

Service: AWS Marketplace Catalog Service

An object that contains metadata and details about the entity.

Contents

Note

In the following list, the required parameters are described first.

DetailsDocument

An object that contains all the details of the entity.

Type: JSON value

Required: No

EntityArn

The Amazon Resource Name (ARN) of the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

Required: No

EntityIdentifier

The ID of the entity, in the format of EntityId@RevisionId.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\\w\\-@]+$`

Required: No

EntityType

The entity type of the entity, in the format of `EntityType@Version`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: No

LastModifiedDate

The last time the entity was modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntityRequest

Service: AWS Marketplace Catalog Service

An object that contains entity ID and the catalog in which the entity is present.

Contents

Note

In the following list, the required parameters are described first.

Catalog

The name of the catalog the entity is present in. The only value at this time is `AWSMarketplace`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: Yes

EntityId

The ID of the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntitySummary

Service: AWS Marketplace Catalog Service

This object is a container for common summary information about the entity. The summary doesn't contain the whole entity structure, but it does contain information common across all entities.

Contents

Note

In the following list, the required parameters are described first.

AmiProductSummary

An object that contains summary information about the AMI product.

Type: [AmiProductSummary](#) object

Required: No

ContainerProductSummary

An object that contains summary information about the container product.

Type: [ContainerProductSummary](#) object

Required: No

DataProductSummary

An object that contains summary information about the data product.

Type: [DataProductSummary](#) object

Required: No

EntityArn

The ARN associated with the unique identifier for the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

Required: No

EntityId

The unique identifier for the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\w\ -]+$`

Required: No

EntityType

The type of the entity.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: No

LastModifiedDate

The last time the entity was published, using ISO 8601 format (2018-02-27T13:45:22Z).

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):(0-5)[\d]):(0-5)[\d]Z$`

Required: No

Name

The name for the entity. This value is not unique. It is defined by the seller.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^\S+[\S\s]*`

Required: No

OfferSummary

An object that contains summary information about the offer.

Type: [OfferSummary](#) object

Required: No

ResaleAuthorizationSummary

An object that contains summary information about the Resale Authorization.

Type: [ResaleAuthorizationSummary](#) object

Required: No

SaaSProductSummary

An object that contains summary information about the SaaS product.

Type: [SaaSProductSummary](#) object

Required: No

Visibility

The visibility status of the entity to buyers. This value can be `Public` (everyone can view the entity), `Limited` (the entity is visible to limited accounts only), or `Restricted` (the entity was published and then unpublished and only existing buyers can view it).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntityTypeFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields per entity type.

Contents

Note

In the following list, the required parameters are described first.

Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

AmiProductFilters

A filter for AMI products.

Type: [AmiProductFilters](#) object

Required: No

ContainerProductFilters

A filter for container products.

Type: [ContainerProductFilters](#) object

Required: No

DataProductFilters

A filter for data products.

Type: [DataProductFilters](#) object

Required: No

OfferFilters

A filter for offers.

Type: [OfferFilters](#) object

Required: No

ResaleAuthorizationFilters

A filter for Resale Authorizations.

Type: [ResaleAuthorizationFilters](#) object

Required: No

SaaSProductFilters

A filter for SaaS products.

Type: [SaaSProductFilters](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntityTypeSort

Service: AWS Marketplace Catalog Service

Object containing all the sort fields per entity type.

Contents

Note

In the following list, the required parameters are described first.

Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

AmiProductSort

A sort for AMI products.

Type: [AmiProductSort](#) object

Required: No

ContainerProductSort

A sort for container products.

Type: [ContainerProductSort](#) object

Required: No

DataProductSort

A sort for data products.

Type: [DataProductSort](#) object

Required: No

OfferSort

A sort for offers.

Type: [OfferSort](#) object

Required: No

ResaleAuthorizationSort

A sort for Resale Authorizations.

Type: [ResaleAuthorizationSort](#) object

Required: No

SaaSProductSort

A sort for SaaS products.

Type: [SaaSProductSort](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ErrorDetail

Service: AWS Marketplace Catalog Service

Details about the error.

Contents

Note

In the following list, the required parameters are described first.

ErrorCode

The error code that identifies the type of error.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 72.

Pattern: `^[a-zA-Z_]+$`

Required: No

ErrorMessage

The message for the error.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Filter

Service: AWS Marketplace Catalog Service

A filter object, used to optionally filter results from calls to the `ListEntities` and `ListChangeSets` actions.

Contents

Note

In the following list, the required parameters are described first.

Name

For `ListEntities`, the supported value for this is an `EntityId`.

For `ListChangeSets`, the supported values are as follows:

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: No

ValueList

`ListEntities` - This is a list of unique `EntityIds`.

`ListChangeSets` - The supported filter names and associated `ValueLists` is as follows:

- `ChangeSetName` - The supported `ValueList` is a list of non-unique `ChangeSetNames`. These are defined when you call the `StartChangeSet` action.
- `Status` - The supported `ValueList` is a list of statuses for all change set requests.
- `EntityId` - The supported `ValueList` is a list of unique `EntityIds`.
- `BeforeStartTime` - The supported `ValueList` is a list of all change sets that started before the filter value.
- `AfterStartTime` - The supported `ValueList` is a list of all change sets that started after the filter value.

- `BeforeEndTime` - The supported `ValueList` is a list of all change sets that ended before the filter value.
- `AfterEndTime` - The supported `ValueList` is a list of all change sets that ended after the filter value.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferAvailabilityEndDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the AvailabilityEndDate of an offer.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on the AvailabilityEndDate of an offer with date range as input.

Type: [OfferAvailabilityEndDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferAvailabilityEndDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on the AvailabilityEndDate of an offer with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on the AvailabilityEndDate of an offer after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on the AvailabilityEndDate of an offer before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferBuyerAccountsFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the BuyerAccounts of an offer.

Contents

Note

In the following list, the required parameters are described first.

WildcardValue

Allows filtering on the BuyerAccounts of an offer with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferEntityIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the entity id of an offer.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on entity id of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for offers entity. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

AvailabilityEndDate

Allows filtering on the `AvailabilityEndDate` of an offer.

Type: [OfferAvailabilityEndDateFilter](#) object

Required: No

BuyerAccounts

Allows filtering on the `BuyerAccounts` of an offer.

Type: [OfferBuyerAccountsFilter](#) object

Required: No

EntityId

Allows filtering on `EntityId` of an offer.

Type: [OfferEntityIdFilter](#) object

Required: No

LastModifiedDate

Allows filtering on the `LastModifiedDate` of an offer.

Type: [OfferLastModifiedDateFilter](#) object

Required: No

Name

Allows filtering on the Name of an offer.

Type: [OfferNameFilter](#) object

Required: No

ProductId

Allows filtering on the ProductId of an offer.

Type: [OfferProductIdFilter](#) object

Required: No

ReleaseDate

Allows filtering on the ReleaseDate of an offer.

Type: [OfferReleaseDateFilter](#) object

Required: No

ResaleAuthorizationId

Allows filtering on the ResaleAuthorizationId of an offer.

Note

Not all offers have a ResaleAuthorizationId. The response will only include offers for which you have permissions.

Type: [OfferResaleAuthorizationIdFilter](#) object

Required: No

State

Allows filtering on the State of an offer.

Type: [OfferStateFilter](#) object

Required: No

Targeting

Allows filtering on the Targeting of an offer.

Type: [OfferTargetingFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the LastModifiedDate of an offer.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on the LastModifiedDate of an offer with date range as input.

Type: [OfferLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on the LastModifiedDate of an offer with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on the LastModifiedDate of an offer after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01)[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on the LastModifiedDate of an offer before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01)[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferNameFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the Name of an offer.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the Name of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 150.

Pattern: $^(.)+\$$

Required: No

WildcardValue

Allows filtering on the Name of an offer with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 150.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferProductIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ProductId of an offer.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ProductId of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferReleaseDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ReleaseDate of an offer.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on the ReleaseDate of an offer with date range as input.

Type: [OfferReleaseDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferReleaseDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on the ReleaseDate of an offer with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on the ReleaseDate of offers after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01)[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on the ReleaseDate of offers before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01)[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferResaleAuthorizationIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ResaleAuthorizationId of an offer.

Note

Not all offers have a ResaleAuthorizationId. The response will only include offers for which you have permissions.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ResaleAuthorizationId of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

OfferSort

Service: AWS Marketplace Catalog Service

Allows to sort offers.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Allows to sort offers.

Type: String

Valid Values: EntityId | Name | ProductId | ResaleAuthorizationId | ReleaseDate | AvailabilityEndDate | BuyerAccounts | State | Targeting | LastModifiedDate

Required: No

SortOrder

Allows to sort offers.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

OfferStateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the State of an offer.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the State of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 2 items.

Valid Values: Draft | Released

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferSummary

Service: AWS Marketplace Catalog Service

Summarized information about an offer.

Contents

Note

In the following list, the required parameters are described first.

AvailabilityEndDate

The availability end date of the offer.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01)[\d]):(0-5)[\d]):(0-5)[\d])Z$`

Required: No

BuyerAccounts

The buyer accounts in the offer.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 26 items.

Length Constraints: Fixed length of 12.

Pattern: `^\d{12}$`

Required: No

Name

The name of the offer.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 150.

Pattern: `^(.)+$`

Required: No

ProductId

The product ID of the offer.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

ReleaseDate

The release date of the offer.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

ResaleAuthorizationId

The ResaleAuthorizationId of the offer.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

State

The status of the offer.

Type: String

Valid Values: Draft | Released

Required: No

Targeting

The targeting in the offer.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 4 items.

Valid Values: BuyerAccounts | ParticipatingPrograms | CountryCodes | None

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OfferTargetingFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the Targeting of an offer.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the Targeting of an offer with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Valid Values: BuyerAccounts | ParticipatingPrograms | CountryCodes | None

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationAvailabilityEndDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on AvailabilityEndDate of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on AvailabilityEndDate of a ResaleAuthorization with date range as input

Type: [ResaleAuthorizationAvailabilityEndDateFilterDateRange](#) object

Required: No

ValueList

Allows filtering on AvailabilityEndDate of a ResaleAuthorization with date value as input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationAvailabilityEndDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on AvailabilityEndDate of a ResaleAuthorization with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on AvailabilityEndDate of a ResaleAuthorization after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on AvailabilityEndDate of a ResaleAuthorization before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationCreatedDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on CreatedDate of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on CreatedDate of a ResaleAuthorization with date range as input.

Type: [ResaleAuthorizationCreatedDateFilterDateRange](#) object

Required: No

ValueList

Allows filtering on CreatedDate of a ResaleAuthorization with date value as input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationCreatedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on CreatedDate of a ResaleAuthorization with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on CreatedDate of a ResaleAuthorization after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on CreatedDate of a ResaleAuthorization before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|01[\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationEntityIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on EntityId of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on EntityId of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for resale authorization entity. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

AvailabilityEndDate

Allows filtering on the `AvailabilityEndDate` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationAvailabilityEndDateFilter](#) object

Required: No

CreatedDate

Allows filtering on the `CreatedDate` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationCreatedDateFilter](#) object

Required: No

EntityId

Allows filtering on the `EntityId` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationEntityIdFilter](#) object

Required: No

LastModifiedDate

Allows filtering on the `LastModifiedDate` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationLastModifiedDateFilter](#) object

Required: No

ManufacturerAccountId

Allows filtering on the `ManufacturerAccountId` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationManufacturerAccountIdFilter](#) object

Required: No

ManufacturerLegalName

Allows filtering on the `ManufacturerLegalName` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationManufacturerLegalNameFilter](#) object

Required: No

Name

Allows filtering on the `Name` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationNameFilter](#) object

Required: No

OfferExtendedStatus

Allows filtering on the `OfferExtendedStatus` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationOfferExtendedStatusFilter](#) object

Required: No

ProductId

Allows filtering on the `ProductId` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationProductIdFilter](#) object

Required: No

ProductName

Allows filtering on the `ProductName` of a `ResaleAuthorization`.

Type: [ResaleAuthorizationProductNameFilter](#) object

Required: No

ResellerAccountID

Allows filtering on the ResellerAccountID of a ResaleAuthorization.

Type: [ResaleAuthorizationResellerAccountIDFilter](#) object

Required: No

ResellerLegalName

Allows filtering on the ResellerLegalName of a ResaleAuthorization.

Type: [ResaleAuthorizationResellerLegalNameFilter](#) object

Required: No

Status

Allows filtering on the Status of a ResaleAuthorization.

Type: [ResaleAuthorizationStatusFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the LastModifiedDate of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

DateRange

Allows filtering on the LastModifiedDate of a ResaleAuthorization with date range as input.

Type: [ResaleAuthorizationLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Allows filtering on the LastModifiedDate of a ResaleAuthorization with date range as input.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Allows filtering on the LastModifiedDate of a ResaleAuthorization after a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Allows filtering on the LastModifiedDate of a ResaleAuthorization before a date.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationManufacturerAccountIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the `ManufacturerAccountId` of a `ResaleAuthorization`.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the `ManufacturerAccountId` of a `ResaleAuthorization` with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Fixed length of 12.

Pattern: `^\d{12}$`

Required: No

WildcardValue

Allows filtering on the `ManufacturerAccountId` of a `ResaleAuthorization` with wild card input.

Type: String

Length Constraints: Fixed length of 12.

Pattern: `^\d{12}$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationManufacturerLegalNameFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the `ManufacturerLegalName` of a `ResaleAuthorization`.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the `ManufacturerLegalName` of a `ResaleAuthorization` with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

WildcardValue

Allows filtering on the `ManufacturerLegalName` of a `ResaleAuthorization` with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationNameFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the Name of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the Name of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

Allows filtering on the Name of a ResaleAuthorization with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationOfferExtendedStatusFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the OfferExtendedStatus of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the OfferExtendedStatus of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationProductIdFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ProductId of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ProductId of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

Allows filtering on the ProductId of a ResaleAuthorization with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationProductNameFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ProductName of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ProductName of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

Allows filtering on the ProductName of a ResaleAuthorization with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationResellerAccountIDFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ResellerAccountID of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ResellerAccountID of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Fixed length of 12.

Pattern: $\backslash d\{12\}$

Required: No

WildcardValue

Allows filtering on the ResellerAccountID of a ResaleAuthorization with wild card input.

Type: String

Length Constraints: Fixed length of 12.

Pattern: $\backslash d\{12\}$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationResellerLegalNameFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the ResellerLegalName of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the ResellerLegalNameProductName of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

Allows filtering on the ResellerLegalName of a ResaleAuthorization with wild card input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationSort

Service: AWS Marketplace Catalog Service

Allows to sort ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Allows to sort ResaleAuthorization.

Type: String

Valid Values: EntityId | Name | ProductId | ProductName |
ManufacturerAccountId | ManufacturerLegalName | ResellerAccountID
| ResellerLegalName | Status | OfferExtendedStatus | CreatedDate |
AvailabilityEndDate | LastModifiedDate

Required: No

SortOrder

Allows to sort ResaleAuthorization.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationStatusFilter

Service: AWS Marketplace Catalog Service

Allows filtering on the Status of a ResaleAuthorization.

Contents

Note

In the following list, the required parameters are described first.

ValueList

Allows filtering on the Status of a ResaleAuthorization with list input.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Valid Values: Draft | Active | Restricted

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResaleAuthorizationSummary

Service: AWS Marketplace Catalog Service

Summarized information about a Resale Authorization.

Contents

Note

In the following list, the required parameters are described first.

AvailabilityEndDate

The availability end date of the ResaleAuthorization.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):(0-5[\d]):(0-5[\d])Z$`

Required: No

CreatedDate

The created date of the ResaleAuthorization.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):(0-5[\d]):(0-5[\d])Z$`

Required: No

ManufacturerAccountId

The manufacturer account ID of the ResaleAuthorization.

Type: String

Length Constraints: Fixed length of 12.

Pattern: `^\d{12}$`

Required: No

ManufacturerLegalName

The manufacturer legal name of the ResaleAuthorization.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Name

The name of the ResaleAuthorization.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

OfferExtendedStatus

The offer extended status of the ResaleAuthorization

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

ProductId

The product ID of the ResaleAuthorization.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

ProductName

The product name of the ResaleAuthorization.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

ResellerAccountID

The reseller account ID of the ResaleAuthorization.

Type: String

Length Constraints: Fixed length of 12.

Pattern: `^\d{12}$`

Required: No

ResellerLegalName

The reseller legal name of the ResaleAuthorization

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Status

The status of the ResaleAuthorization.

Type: String

Valid Values: Draft | Active | Restricted

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductEntityIdFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on entity id of a SaaS product.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique entity id values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z0-9][.a-zA-Z0-9/-]+[a-zA-Z0-9]$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductFilters

Service: AWS Marketplace Catalog Service

Object containing all the filter fields for SaaS products. Client can add only one wildcard filter and a maximum of 8 filters in a single `ListEntities` request.

Contents

Note

In the following list, the required parameters are described first.

EntityId

Unique identifier for the SaaS product.

Type: [SaaSProductEntityIdFilter](#) object

Required: No

LastModifiedDate

The last date on which the SaaS product was modified.

Type: [SaaSProductLastModifiedDateFilter](#) object

Required: No

ProductTitle

The title of the SaaS product.

Type: [SaaSProductTitleFilter](#) object

Required: No

Visibility

The visibility of the SaaS product.

Type: [SaaSProductVisibilityFilter](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductLastModifiedDateFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering based on the last modified date of SaaS products

Contents

Note

In the following list, the required parameters are described first.

DateRange

Dates between which the SaaS product was last modified.

Type: [SaaSProductLastModifiedDateFilterDateRange](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductLastModifiedDateFilterDateRange

Service: AWS Marketplace Catalog Service

Object that contains date range of the last modified date to be filtered on. You can optionally provide a `BeforeValue` and/or `AfterValue`. Both are inclusive.

Contents

Note

In the following list, the required parameters are described first.

AfterValue

Date after which the SaaS product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

BeforeValue

Date before which the SaaS product was last modified.

Type: String

Length Constraints: Fixed length of 20.

Pattern: `^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|[12][\d])T(2[0-3]|[01][\d]):([0-5][\d]):([0-5][\d])Z$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductSort

Service: AWS Marketplace Catalog Service

Objects that allows sorting on SaaS products based on certain fields and sorting order.

Contents

Note

In the following list, the required parameters are described first.

SortBy

Field to sort the SaaS products by.

Type: String

Valid Values: EntityId | ProductTitle | Visibility | LastModifiedDate

Required: No

SortOrder

The sorting order. Can be ASCENDING or DESCENDING. The default value is DESCENDING.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductSummary

Service: AWS Marketplace Catalog Service

Object that contains summarized information about a SaaS product.

Contents

Note

In the following list, the required parameters are described first.

ProductTitle

The title of the SaaS product.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^(.)+$`

Required: No

Visibility

The lifecycle of the SaaS product.

Type: String

Valid Values: `Limited` | `Public` | `Restricted` | `Draft`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

SaaSProductTitleFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on product title.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique product title values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

WildcardValue

A string that will be the `wildCard` input for product tile filter. It matches the provided value as a substring in the actual value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SaaSProductVisibilityFilter

Service: AWS Marketplace Catalog Service

Object that allows filtering on the visibility of the product in the AWS Marketplace.

Contents

Note

In the following list, the required parameters are described first.

ValueList

A string array of unique visibility values to be filtered on.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Valid Values: Limited | Public | Restricted | Draft

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Sort

Service: AWS Marketplace Catalog Service

An object that contains two attributes, `SortBy` and `SortOrder`.

Contents

Note

In the following list, the required parameters are described first.

SortBy

For `ListEntities`, supported attributes include `LastModifiedDate` (default) and `EntityId`. In addition to `LastModifiedDate` and `EntityId`, each `EntityType` might support additional fields.

For `ListChangeSets`, supported attributes include `StartTime` and `EndTime`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[a-zA-Z]+$`

Required: No

SortOrder

The sorting order. Can be `ASCENDING` or `DESCENDING`. The default value is `DESCENDING`.

Type: String

Valid Values: `ASCENDING` | `DESCENDING`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Tag

Service: AWS Marketplace Catalog Service

A list of objects specifying each key name and value.

Contents

Note

In the following list, the required parameters are described first.

Key

The key associated with the tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: Yes

Value

The value associated with the tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Agreement API

The following data types are supported by AWS Marketplace Agreement API:

- [AcceptedTerm](#)
- [Acceptor](#)
- [AgreementViewSummary](#)
- [ByolPricingTerm](#)
- [ConfigurableUpfrontPricingTerm](#)
- [ConfigurableUpfrontPricingTermConfiguration](#)
- [ConfigurableUpfrontRateCardItem](#)
- [Constraints](#)
- [Dimension](#)
- [DocumentItem](#)
- [EstimatedCharges](#)
- [Filter](#)
- [FixedUpfrontPricingTerm](#)
- [FreeTrialPricingTerm](#)
- [GrantItem](#)
- [LegalTerm](#)
- [PaymentScheduleTerm](#)
- [ProposalSummary](#)
- [Proposer](#)
- [RateCardItem](#)
- [RecurringPaymentTerm](#)
- [RenewalTerm](#)
- [RenewalTermConfiguration](#)
- [Resource](#)

- [ScheduleItem](#)
- [Selector](#)
- [Sort](#)
- [SupportTerm](#)
- [UsageBasedPricingTerm](#)
- [UsageBasedRateCardItem](#)
- [ValidationExceptionField](#)
- [ValidityTerm](#)

AcceptedTerm

Service: AWS Marketplace Agreement Service

A subset of terms proposed by the proposer, which have been accepted by the acceptor as part of agreement creation.

Contents

Note

In the following list, the required parameters are described first.

Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

byolPricingTerm

Enables you and your customers to move your existing agreements to AWS Marketplace. The customer won't be charged for product usage in AWS Marketplace because they already paid for the product outside of AWS Marketplace.

Type: [ByolPricingTerm](#) object

Required: No

configurableUpfrontPricingTerm

Defines a prepaid payment model that allows buyers to configure the entitlements they want to purchase and the duration.

Type: [ConfigurableUpfrontPricingTerm](#) object

Required: No

fixedUpfrontPricingTerm

Defines a pre-paid pricing model where the customers are charged a fixed upfront amount.

Type: [FixedUpfrontPricingTerm](#) object

Required: No

freeTrialPricingTerm

Defines a short-term free pricing model where the buyers aren't charged anything within a specified limit.

Type: [FreeTrialPricingTerm](#) object

Required: No

legalTerm

Defines the list of text agreements proposed to the acceptors. An example is the end user license agreement (EULA).

Type: [LegalTerm](#) object

Required: No

paymentScheduleTerm

Defines an installment-based pricing model where customers are charged a fixed price on different dates during the agreement validity period. This is used most commonly for flexible payment schedule pricing.

Type: [PaymentScheduleTerm](#) object

Required: No

recurringPaymentTerm

Defines a pricing model where customers are charged a fixed recurring price at the end of each billing period.

Type: [RecurringPaymentTerm](#) object

Required: No

renewalTerm

Defines that on graceful expiration of the agreement (when the agreement ends on its pre-defined end date), a new agreement will be created using the accepted terms on the existing agreement. In other words, the agreement will be renewed. Presence of `RenewalTerm` in the

offer document means that auto-renewal is allowed. Buyers will have the option to accept or decline auto-renewal at the offer acceptance/agreement creation. Buyers can also change this flag from `True` to `False` or `False` to `True` at anytime during the agreement's lifecycle.

Type: [RenewalTerm](#) object

Required: No

supportTerm

Defines the customer support available for the acceptors when they purchase the software.

Type: [SupportTerm](#) object

Required: No

usageBasedPricingTerm

Defines a usage-based pricing model (typically, pay-as-you-go pricing), where the customers are charged based on product usage.

Type: [UsageBasedPricingTerm](#) object

Required: No

validityTerm

Defines the conditions that will keep an agreement created from this offer valid.

Type: [ValidityTerm](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Acceptor

Service: AWS Marketplace Agreement Service

The details of the party accepting the agreement terms. This is commonly the buyer for PurchaseAgreement.

Contents

Note

In the following list, the required parameters are described first.

accountId

The AWS account ID of the acceptor.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `^[0-9]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AgreementViewSummary

Service: AWS Marketplace Agreement Service

A summary of the agreement, including top-level attributes (for example, the agreement ID, version, proposer, and acceptor).

Contents

Note

In the following list, the required parameters are described first.

acceptanceTime

The date and time that the agreement was accepted.

Type: Timestamp

Required: No

acceptor

Details of the party accepting the agreement terms. This is commonly the buyer for `PurchaseAgreement`.

Type: [Acceptor](#) object

Required: No

agreementId

The unique identifier of the agreement.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: No

agreementType

The type of agreement. Values are `PurchaseAgreement` or `VendorInsightsAgreement`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z]+$`

Required: No

endTime

The date and time when the agreement ends. The field is `null` for pay-as-you-go agreements, which don't have end dates.

Type: Timestamp

Required: No

proposalSummary

A summary of the proposal

Type: [ProposalSummary](#) object

Required: No

proposer

Details of the party proposing the agreement terms, most commonly the seller for `PurchaseAgreement`.

Type: [Proposer](#) object

Required: No

startTime

The date and time when the agreement starts.

Type: Timestamp

Required: No

status

The current status of the agreement.

Type: String

Valid Values: ACTIVE | ARCHIVED | CANCELLED | EXPIRED | RENEWED | REPLACED | ROLLED_BACK | SUPERSEDED | TERMINATED

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ByolPricingTerm

Service: AWS Marketplace Agreement Service

Enables you and your customers to move your existing agreements to AWS Marketplace. The customer won't be charged for product usage in AWS Marketplace because they already paid for the product outside of AWS Marketplace.

Contents

Note

In the following list, the required parameters are described first.

type

Type of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ConfigurableUpfrontPricingTerm

Service: AWS Marketplace Agreement Service

Defines a prepaid payment model that allows buyers to configure the entitlements they want to purchase and the duration.

Contents

Note

In the following list, the required parameters are described first.

configuration

Additional parameters specified by the acceptor while accepting the term.

Type: [ConfigurableUpfrontPricingTermConfiguration](#) object

Required: No

currencyCode

Defines the currency for the prices mentioned in the term.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]+\$$

Required: No

rateCards

A rate card defines the per unit rates for product dimensions.

Type: Array of [ConfigurableUpfrontRateCardItem](#) objects

Required: No

type

Category of selector.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ConfigurableUpfrontPricingTermConfiguration

Service: AWS Marketplace Agreement Service

Defines a prepaid payment model that allows buyers to configure the entitlements they want to purchase and the duration.

Contents

Note

In the following list, the required parameters are described first.

dimensions

Defines the dimensions that the acceptor has purchased from the overall set of dimensions presented in the rate card.

Type: Array of [Dimension](#) objects

Array Members: Minimum number of 1 item.

Required: Yes

selectorValue

Defines the length of time for which the particular pricing/dimension is being purchased by the acceptor.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ConfigurableUpfrontRateCardItem

Service: AWS Marketplace Agreement Service

Within the prepaid payment model defined under `ConfigurableUpfrontPricingTerm`, the `RateCardItem` defines all the various rate cards (including pricing and dimensions) that have been proposed.

Contents

Note

In the following list, the required parameters are described first.

constraints

Defines limits on how the term can be configured by acceptors.

Type: [Constraints](#) object

Required: No

rateCard

Defines the per unit rates for product dimensions.

Type: Array of [RateCardItem](#) objects

Required: No

selector

Differentiates between the mutually exclusive rate cards in the same pricing term to be selected by the buyer.

Type: [Selector](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Constraints

Service: AWS Marketplace Agreement Service

Defines limits on how the term can be configured by acceptors.

Contents

Note

In the following list, the required parameters are described first.

multipleDimensionSelection

Determines if buyers are allowed to select multiple dimensions in the rate card. The possible values are `Allowed` and `Disallowed`. The default value is `Allowed`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

quantityConfiguration

Determines if acceptors are allowed to configure quantity for each dimension in rate card. The possible values are `Allowed` and `Disallowed`. The default value is `Allowed`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Dimension

Service: AWS Marketplace Agreement Service

Defines the dimensions that the acceptor has purchased from the overall set of dimensions presented in the rate card.

Contents

Note

In the following list, the required parameters are described first.

dimensionKey

The name of key value of the dimension.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: Yes

dimensionValue

The number of units of the dimension the acceptor has purchased.

Note

For Agreements with `ConfigurableUpfrontPricingTerm`, the `RateCard` section will define the prices and dimensions defined by the seller (proposer), whereas the `Configuration` section will define the actual dimensions, prices, and units the buyer has chosen to accept.

Type: Integer

Valid Range: Minimum value of 0.

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DocumentItem

Service: AWS Marketplace Agreement Service

Includes the list of references to legal resources proposed by the proposer to the acceptor. Each DocumentItem refers to an individual reference.

Contents

Note

In the following list, the required parameters are described first.

type

Category of the document. Document types include:

- `CustomEula` – A custom EULA provided by you as seller. A URL for a EULA stored in an accessible Amazon S3 bucket is required for this document type.
- `CustomDsa` – A custom Data Subscription Agreement (DSA) provided by you as seller. A URL for a DSA stored in an accessible Amazon S3 bucket is required for this document type.
- `StandardEula` – The Standard Contract for AWS Marketplace (SCMP). For more information about SCMP, see the AWS Marketplace Seller Guide. You don't provide a URL for this type because it's managed by AWS Marketplace.
- `StandardDsa` – DSA for AWS Marketplace. For more information about the DSA, see the AWS Data Exchange User Guide. You don't provide a URL for this type because it's managed by AWS Marketplace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

url

A URL to the legal document for buyers to read. Required when Type is `CustomEula`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

version

Version of standard contracts provided by AWS Marketplace. Required when Type is `StandardEula` or `StandardDsa`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EstimatedCharges

Service: AWS Marketplace Agreement Service

Estimated cost of the agreement.

Contents

Note

In the following list, the required parameters are described first.

agreementValue

The total known amount customer has to pay across the lifecycle of the agreement.

Note

This is the total contract value if accepted terms contain `ConfigurableUpfrontPricingTerm` or `FixedUpfrontPricingTerm`. In the case of pure contract pricing, this will be the total value of the contract. In the case of contracts with consumption pricing, this will only include the committed value and not include any overages that occur.

If the accepted terms contain `PaymentScheduleTerm`, it will be the total payment schedule amount. This occurs when flexible payment schedule is used, and is the sum of all invoice charges in the payment schedule.

In case a customer has amended an agreement, by purchasing more units of any dimension, this will include both the original cost as well as the added cost incurred due to addition of new units.

This is 0 if the accepted terms contain `UsageBasedPricingTerm` without `ConfigurableUpfrontPricingTerm` or `RecurringPaymentTerm`. This occurs for usage-based pricing (such as SaaS metered or AMI/container hourly or monthly), because the exact usage is not known upfront.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

currencyCode

Defines the currency code for the charge.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Filter

Service: AWS Marketplace Agreement Service

The filter name and value pair that is used to return a more specific list of results. Filters can be used to match a set of resources by various criteria, such as `offerId` or `productId`.

Contents

Note

In the following list, the required parameters are described first.

name

The name of the filter.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `^[A-Za-z_]+$`

Required: No

values

The filter value.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9+:_-]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FixedUpfrontPricingTerm

Service: AWS Marketplace Agreement Service

Defines a prepaid pricing model where the customers are charged a fixed upfront amount.

Contents

Note

In the following list, the required parameters are described first.

currencyCode

Defines the currency for the prices mentioned in this term.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]+\$$

Required: No

duration

Contract duration for the terms.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

grants

Entitlements granted to the acceptor of fixed upfront as part of agreement execution.

Type: Array of [GrantItem](#) objects

Required: No

price

Fixed amount to be charged to the customer when this term is accepted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

type

Category of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^[A-Za-z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FreeTrialPricingTerm

Service: AWS Marketplace Agreement Service

Defines a short-term free pricing model where the buyers aren't charged anything within a specified limit.

Contents

Note

In the following list, the required parameters are described first.

duration

Duration of the free trial period (5–31 days).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

grants

Entitlements granted to the acceptor of a free trial as part of an agreement execution.

Type: Array of [GrantItem](#) objects

Required: No

type

Category of the term.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^[A-Za-z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

GrantItem

Service: AWS Marketplace Agreement Service

Entitlements granted to the acceptor of fixed upfront as part of agreement execution.

Contents

Note

In the following list, the required parameters are described first.

dimensionKey

Unique dimension key defined in the product document. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

maxQuantity

Maximum amount of capacity that the buyer can be entitled to the given dimension of the product. If MaxQuantity is not provided, the buyer will be able to use an unlimited amount of the given dimension.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

LegalTerm

Service: AWS Marketplace Agreement Service

Defines the list of text agreements proposed to the acceptors. An example is the end user license agreement (EULA).

Contents

Note

In the following list, the required parameters are described first.

documents

List of references to legal resources proposed to the buyers. An example is the EULA.

Type: Array of [DocumentItem](#) objects

Required: No

type

Category of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

PaymentScheduleTerm

Service: AWS Marketplace Agreement Service

Defines an installment-based pricing model where customers are charged a fixed price on different dates during the agreement validity period. This is used most commonly for flexible payment schedule pricing.

Contents

Note

In the following list, the required parameters are described first.

currencyCode

Defines the currency for the prices mentioned in the term.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]^+$

Required: No

schedule

List of the payment schedule where each element defines one installment of payment. It contains the information necessary for calculating the price.

Type: Array of [ScheduleItem](#) objects

Required: No

type

Type of the term.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProposalSummary

Service: AWS Marketplace Agreement Service

A summary of the proposal received from the proposer.

Contents

Note

In the following list, the required parameters are described first.

offerId

The unique identifier of the offer in AWS Marketplace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^\S{1,64}$`

Required: No

resources

The list of resources involved in the agreement.

Type: Array of [Resource](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Proposer

Service: AWS Marketplace Agreement Service

Details of the party proposing the agreement terms,. This is commonly the seller for PurchaseAgreement.

Contents

Note

In the following list, the required parameters are described first.

accountId

The AWS account ID of the proposer.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `^[0-9]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RateCardItem

Service: AWS Marketplace Agreement Service

Defines the per unit rates for each individual product dimension.

Contents

Note

In the following list, the required parameters are described first.

dimensionKey

Dimension for which the given entitlement applies. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

price

Per unit price for the product dimension that's used for calculating the amount to be charged.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RecurringPaymentTerm

Service: AWS Marketplace Agreement Service

Defines a pricing model where customers are charged a fixed recurring price at the end of each billing period.

Contents

Note

In the following list, the required parameters are described first.

billingPeriod

Defines the recurrence at which buyers are charged.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

currencyCode

Defines the currency for the prices mentioned in this term.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]+\$$

Required: No

price

Amount charged to the buyer every billing period.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

type

Type of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^[A-Za-z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RenewalTerm

Service: AWS Marketplace Agreement Service

Defines that on graceful expiration of the agreement (when the agreement ends on its pre-defined end date), a new agreement will be created using the accepted terms on the existing agreement. In other words, the agreement will be renewed. The presence of `RenewalTerm` in the offer document means that auto-renewal is allowed. Buyers will have the option to accept or decline auto-renewal at the offer acceptance/agreement creation. Buyers can also change this flag from `True` to `False` or `False` to `True` at anytime during the agreement's lifecycle.

Contents

Note

In the following list, the required parameters are described first.

configuration

Additional parameters specified by the acceptor while accepting the term.

Type: [RenewalTermConfiguration](#) object

Required: No

type

Category of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RenewalTermConfiguration

Service: AWS Marketplace Agreement Service

Additional parameters specified by the acceptor while accepting the term.

Contents

Note

In the following list, the required parameters are described first.

enableAutoRenew

Defines whether the acceptor has chosen to auto-renew the agreement at the end of its lifecycle. Can be set to `True` or `False`.

Type: Boolean

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Resource

Service: AWS Marketplace Agreement Service

The list of resources involved in the agreement.

Contents

Note

In the following list, the required parameters are described first.

id

The unique identifier of the resource.

Note

We mention the term resource, which is most commonly a product, so a `resourceId` is also a `productId`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[A-Za-z0-9_/-]+$`

Required: No

type

Type of the resource, which is the product. Values include `SaaSProduct` or `AmiProduct`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ScheduleItem

Service: AWS Marketplace Agreement Service

An individual installment of the payment that includes the date and amount of the charge.

Contents

Note

In the following list, the required parameters are described first.

chargeAmount

The price that the customer would pay on the scheduled date (chargeDate).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

chargeDate

The date that the customer would pay the price defined in this payment schedule term. Invoices are generated on the date provided.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Selector

Service: AWS Marketplace Agreement Service

Differentiates between the mutually exclusive rate cards in the same pricing term to be selected by the buyer.

Contents

Note

In the following list, the required parameters are described first.

type

Category of selector.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

value

Contract duration. This field supports the ISO 8601 format.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Sort

Service: AWS Marketplace Agreement Service

An object that contains the `SortBy` and `SortOrder` attributes.

Contents

Note

In the following list, the required parameters are described first.

sortBy

The attribute on which the data is grouped, which can be by `StartTime` and `EndTime`. The default value is `EndTime`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[A-Za-z_]+$`

Required: No

sortOrder

The sorting order, which can be `ASCENDING` or `DESCENDING`. The default value is `DESCENDING`.

Type: String

Valid Values: `ASCENDING` | `DESCENDING`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SupportTerm

Service: AWS Marketplace Agreement Service

Defines the customer support available for the acceptors when they purchase the software.

Contents

Note

In the following list, the required parameters are described first.

refundPolicy

Free-text field about the refund policy description that will be shown to customers as is on the website and console.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: No

type

Category of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^[A-Za-z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UsageBasedPricingTerm

Service: AWS Marketplace Agreement Service

Defines a usage-based pricing model (typically, pay-as-you-go pricing), where the customers are charged based on product usage.

Contents

Note

In the following list, the required parameters are described first.

currencyCode

Defines the currency for the prices mentioned in the term.

Type: String

Length Constraints: Fixed length of 3.

Pattern: $^[A-Z]+\$$

Required: No

rateCards

List of rate cards.

Type: Array of [UsageBasedRateCardItem](#) objects

Required: No

type

Category of the term.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^[A-Za-z]+\$$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UsageBasedRateCardItem

Service: AWS Marketplace Agreement Service

Within the pay-as-you-go model defined under `UsageBasedPricingTerm`, the `UsageBasedRateCardItem` defines an individual rate for a product dimension.

Contents

Note

In the following list, the required parameters are described first.

`rateCard`

Defines the per unit rates for product dimensions.

Type: Array of [RateCardItem](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ValidationExceptionField

Service: AWS Marketplace Agreement Service

The input fails to satisfy the constraints specified by the service.

Contents

Note

In the following list, the required parameters are described first.

message

See applicable actions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: Yes

name

The name of the field associated with the error.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: $^(.)+\$$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ValidityTerm

Service: AWS Marketplace Agreement Service

Defines the conditions that will keep an agreement created from this offer valid.

Contents

Note

In the following list, the required parameters are described first.

agreementDuration

Defines the duration that the agreement remains active. If `AgreementStartDate` isn't provided, the agreement duration is relative to the agreement signature time. The duration is represented in the ISO_8601 format.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^(.)+$`

Required: No

agreementEndDate

Defines the date when the agreement ends. The agreement ends at 23:59:59.999 UTC on the date provided. If `AgreementEndDate` isn't provided, the agreement end date is determined by the validity of individual terms.

Type: Timestamp

Required: No

agreementStartDate

Defines the date when agreement starts. The agreement starts at 00:00:00.000 UTC on the date provided. If `AgreementStartDate` isn't provided, the agreement start date is determined based on agreement signature time.

Type: Timestamp

Required: No

type

Category of the term being updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Pattern: `^[A-Za-z]+$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Metering API

The following data types are supported by AWS Marketplace Metering API:

- [Tag](#)
- [UsageAllocation](#)
- [UsageRecord](#)
- [UsageRecordResult](#)

Tag

Service: AWSMarketplace Metering

Metadata assigned to an allocation. Each tag is made up of a key and a value.

Contents

Note

In the following list, the required parameters are described first.

Key

One part of a key-value pair that makes up a tag. A key is a label that acts like a category for the specific tag values.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^[a-zA-Z0-9+ - = . _ : \ / @] + $`

Required: Yes

Value

One part of a key-value pair that makes up a tag. A value acts as a descriptor within a tag category (key). The value can be empty or null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[a-zA-Z0-9+ - = . _ : \ / @] + $`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UsageAllocation

Service: AWSMarketplace Metering

Usage allocations allow you to split usage into buckets by tags.

Each UsageAllocation indicates the usage quantity for a specific set of tags.

Contents

Note

In the following list, the required parameters are described first.

AllocatedUsageQuantity

The total quantity allocated to this bucket of usage.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

Required: Yes

Tags

The set of tags that define the bucket of usage. For the bucket of items with no tags, this parameter can be left out.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UsageRecord

Service: AWSMarketplace Metering

A UsageRecord indicates a quantity of usage for a given product, customer, dimension and time.

Multiple requests with the same UsageRecords as input will be de-duplicated to prevent double charges.

Contents

Note

In the following list, the required parameters are described first.

CustomerIdentifier

The CustomerIdentifier is obtained through the ResolveCustomer operation and represents an individual buyer in your application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[\s\S]+`

Required: Yes

Dimension

During the process of registering a product on AWS Marketplace, dimensions are specified. These represent different units of value in your application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[\s\S]+`

Required: Yes

Timestamp

Timestamp, in UTC, for which the usage is being reported.

Your application can meter usage for up to one hour in the past. Make sure the timestamp value is not before the start of the software usage.

Type: Timestamp

Required: Yes

Quantity

The quantity of usage consumed by the customer for the given dimension and time. Defaults to 0 if not specified.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

Required: No

UsageAllocations

The set of UsageAllocations to submit. The sum of all UsageAllocation quantities must equal the Quantity of the UsageRecord.

Type: Array of [UsageAllocation](#) objects

Array Members: Minimum number of 1 item. Maximum number of 2500 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UsageRecordResult

Service: AWSMarketplace Metering

A `UsageRecordResult` indicates the status of a given `UsageRecord` processed by `BatchMeterUsage`.

Contents

Note

In the following list, the required parameters are described first.

MeteringRecordId

The `MeteringRecordId` is a unique identifier for this metering event.

Type: String

Required: No

Status

The `UsageRecordResult` `Status` indicates the status of an individual `UsageRecord` processed by `BatchMeterUsage`.

- *Success*- The `UsageRecord` was accepted and honored by `BatchMeterUsage`.
- *CustomerNotSubscribed*- The `CustomerIdentifier` specified is not able to use your product. The `UsageRecord` was not honored. There are three causes for this result:
 - The customer identifier is invalid.
 - The customer identifier provided in the metering record does not have an active agreement or subscription with this product. Future `UsageRecords` for this customer will fail until the customer subscribes to your product.
 - The customer's AWS account was suspended.
- *DuplicateRecord*- Indicates that the `UsageRecord` was invalid and not honored. A previously metered `UsageRecord` had the same customer, dimension, and time, but a different quantity.

Type: String

Valid Values: Success | CustomerNotSubscribed | DuplicateRecord

Required: No

UsageRecord

The UsageRecord that was part of the BatchMeterUsage request.

Type: [UsageRecord](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Entitlement API

The following data types are supported by AWS Marketplace Entitlement API:

- [Entitlement](#)
- [EntitlementValue](#)

Entitlement

Service: AWS Marketplace Entitlement Service

An entitlement represents capacity in a product owned by the customer. For example, a customer might own some number of users or seats in an SaaS application or some amount of data capacity in a multi-tenant database.

Contents

Note

In the following list, the required parameters are described first.

CustomerIdentifier

The customer identifier is a handle to each unique customer in an application. Customer identifiers are obtained through the ResolveCustomer operation in AWS Marketplace Metering Service.

Type: String

Pattern: \S+

Required: No

Dimension

The dimension for which the given entitlement applies. Dimensions represent categories of capacity in a product and are specified when the product is listed in AWS Marketplace.

Type: String

Pattern: \S+

Required: No

ExpirationDate

The expiration date represents the minimum date through which this entitlement is expected to remain valid. For contractual products listed on AWS Marketplace, the expiration date is the

date at which the customer will renew or cancel their contract. Customers who are opting to renew their contract will still have entitlements with an expiration date.

Type: Timestamp

Required: No

ProductCode

The product code for which the given entitlement applies. Product codes are provided by AWS Marketplace when the product listing is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: No

Value

The EntitlementValue represents the amount of capacity that the customer is entitled to for the product.

Type: [EntitlementValue](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EntitlementValue

Service: AWS Marketplace Entitlement Service

The EntitlementValue represents the amount of capacity that the customer is entitled to for the product.

Contents

Note

In the following list, the required parameters are described first.

BooleanValue

The BooleanValue field will be populated with a boolean value when the entitlement is a boolean type. Otherwise, the field will not be set.

Type: Boolean

Required: No

DoubleValue

The DoubleValue field will be populated with a double value when the entitlement is a double type. Otherwise, the field will not be set.

Type: Double

Required: No

IntegerValue

The IntegerValue field will be populated with an integer value when the entitlement is an integer type. Otherwise, the field will not be set.

Type: Integer

Required: No

StringValue

The StringValue field will be populated with a string value when the entitlement is a string type. Otherwise, the field will not be set.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Deployment API

The following data types are supported by AWS Marketplace Deployment API:

- [DeploymentParameterInput](#)

DeploymentParameterInput

Service: AWS Marketplace Deployment Service

The shape containing the requested deployment parameter name and secretString.

Note

To support AWS CloudFormation dynamic references to this resource using Quick Launch, this value must match a parameter defined in the CloudFormation templated provided to buyers.

Contents

Note

In the following list, the required parameters are described first.

name

The desired name of the deployment parameter. This is the identifier on which deployment parameters are keyed for a given buyer and product. If this name matches an existing deployment parameter, this request will update the existing resource.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 400.

Pattern: `^[a-zA-Z0-9/_+=.@-]+$`

Required: Yes

secretString

The text to encrypt and store in the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 15000.

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AWS Marketplace Reporting API

The AWS Marketplace Reporting API has no separate data types.

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see [Signing AWS API requests](#) in the *IAM User Guide*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key/YYYYMMDD/region/service/aws4_request*.

For more information, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Elements of an AWS API request signature](#) in the *IAM User Guide*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see [AWS services that work with IAM](#) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

Document history

The following table describes the documentation for this release of this API reference.

Change	Description	Date
AWS Marketplace Catalog API topic updates	Updated CPPO prerequisites.	May 9, 2024
AWS Marketplace Catalog API topic updates	Updated instances of the <code>Details</code> attribute to <code>DetailsDocument</code> .	April 30, 2024
AWS Marketplace Catalog API topic updates	Added notes to Working with offers to clarify constraints.	April 17, 2024
AWS Marketplace Catalog API supports service-linked for resale authorization	Updated resale authorization prerequisites for service-linked role.	March 20, 2024
AWS Marketplace Catalog API supports organization units	Added content to enable private marketplace support at Organization Unit (OU) level.	February 16, 2024
AWS Marketplace Catalog API supports setting intent on requests	Sellers now can request changes for entities with specific intent using the AWS Marketplace Catalog API.	February 9, 2024
AWS Marketplace Catalog API supports wildcard filter validation	Added wildcard filter validation in <code>ListEntities</code> API.	February 5, 2024
AWS Marketplace Catalog API supports Amazon EKS add-ons	Added content and error messages related to publishing to Amazon EKS add-ons	January 29, 2024

from AWS Marketplace container-based product.

[AWS Marketplace Catalog API supports listing details about entities](#)

Sellers can now list details about entities using the AWS Marketplace Catalog API.

December 19, 2023

[The AWS Marketplace Deployment Service API reference is now generally available](#)

This service provides an API interface that supports a secure method for passing deployment parameters (for example, API keys and external IDs) during the Quick Launch experience.

November 29, 2023

[The AWS Marketplace Agreement Service API reference is now generally available](#)

This service provides an API interface that helps AWS Marketplace sellers manage their product-related agreements, including listing, searching, and filtering agreements.

November 29, 2023

[AWS Marketplace Catalog API supports the ability to create products, offers, Resale Authorizations, and CPPOs](#)

Sellers can now use the AWS Marketplace Catalog API to create and update [products](#), [offers](#), [Resale Authorizations](#), and [channel partner private offers \(CPPOs\)](#).

November 29, 2023

[AWS Marketplace Catalog API supports enhanced filtering and sorting capabilities](#)

Sellers can now sort and filter products using the AWS Marketplace Catalog API.

November 29, 2023

AWS Marketplace Catalog API supports resource sharing	The AWS Marketplace Catalog API integrates with AWS Resource Access Manager (AWS RAM) to enable resource sharing. See Working with AWS RAM to share resources .	April 12, 2023
AWS Marketplace Discovery API topic update	The AWS Marketplace Discovery API now supports CloudTrail. See Logging AWS Marketplace Discovery API calls using AWS CloudTrail .	December 15, 2022
AWS Marketplace supports archiving private marketplace experiences	Buyers can now archive and reactivate private marketplace experiences in AWS Marketplace. See Working with a private marketplace .	December 12, 2022
AWS Marketplace Private marketplace granular permissions	Buyers now have more granular permissions to manage private marketplace experiences. See Working with a private marketplace .	September 8, 2022
AWS Marketplace Discovery API release notes	Added Release notes for the AWS Marketplace Discovery API.	May 20, 2022
AWS Marketplace Discovery API topic update	Documentation-only update to the AWS Marketplace Discovery API topic .	January 14, 2022

Support for Helm chart delivery options and QuickLaunch for container-based products	Added documentation for adding or updating Helm chart delivery options in container-based product versions, including enabling QuickLaunch for buyers. See Working with container-based products .	November 29, 2021
Support for managing seller products	Added the ability to manage AMI and container products programmatically. See Working with seller products .	March 26, 2021
Support for managing private marketplaces	Added the ability to create and maintain private marketplaces for AWS Organizations programmatically. See Working with a private marketplace .	December 3, 2020
The AWS Marketplace Discovery API is now available	The Discovery API provides programmatic access to find products in the AWS Marketplace. For details, see Discovery API .	September 30, 2020
The AWS Marketplace Catalog API is now generally available	This service provides an API interface for approved providers to programmatically access the self-service publishing capabilities on the AWS Marketplace Management Portal.	November 12, 2019

Release notes for AWS Marketplace Discovery API

Release notes for AWS Marketplace Discovery API. Details about new features, improvements, fixes, and announcements.

Discovery API release notes for 2024

August 15, 2024

Discovery API customers can access the updated Discovery API documentation and SDK on the Amazon Simple Storage Service bucket provided by the Discovery API team. Customers can refer to the Change Log in the private documentation for more details.

Launch announcements

The Discovery API Private SDK is now available in JavaScript V3.

- For information about using the AWS SDK for JavaScript V3, see the [AWS SDK for JavaScript V3 Developer Guide](#).
- For information about migrating from V2 to V3, see [Migrate from version 2.x to 3.x of the AWS SDK for JavaScript](#), in the *AWS SDK for JavaScript V3 Developer Guide*.

Improvements

- AWS SDK paginators are now available for the `SearchListings` and `ListListingViewQueries` API operations in all supported languages that have native SDK support for paginators.

You can still use `NextToken` to manually paginate. You can use `NextToken` with the `SearchListings`, `ListListingViewQueries`, and `GetSearchFacets` API operations across all the SDKs that we vend.

- SDK updated for all supported languages with the latest AWS SDK artifacts.
- Private documentation updated to include an SDK usage section for JavaScript V3.

Discovery API release notes for 2022

May 20, 2022

Published on May 20, 2022

Discovery API customers can access the updated Discovery API documentation and SDK on the Amazon Simple Storage Service (Amazon S3) bucket that the Discovery API team shared with them previously. Refer to the Change Log in the private documentation for more details.

Discovery API announces the following launch, and improvements:

Launch announcements

- Discovery API launched in two additional AWS Regions:
 - US West (Oregon) – us-west-2
 - Europe (Ireland) – eu-west-1
- Discovery API Private SDK is now available in Java 2.x:
 - For more information about how to use the AWS SDK for Java 2.x, see the [AWS SDK for Java 2.x Developer Guide](#).
 - For more information about migration, see [migrating from version 1.x to 2.x of the AWS SDK for Java](#) in the *AWS SDK for Java 2.x Developer Guide*.
 - For more information about changes between versions 1.11.x and 2.x of the AWS SDK for Java 2.x, see [1.11 to 2.x Changelog](#) on the GitHub website.

Improvements

- Enhanced sorting functionality for the SearchListings API operation by introducing new options for:
 - `SortBy` – AVERAGE_CUSTOMER_RATING, CREATION_TIME, LAST_MODIFIED_TIME
 - `SortOrder` – ASCENDING
- SDK updated for all the existing languages with the latest AWS SDK artifacts.
- Documentation updated to include SDK usage section for all languages.