

\*\*\*Unable to locate subtitle\*\*\*

# AWS Marketplace Agreement Service



# **AWS Marketplace Agreement Service: \*\*\*Unable to locate subtitle\*\*\***

# Table of Contents

<b>AWS Marketplace Agreement Service .....</b>	<b>1</b>
Supported AWS Regions .....	1
Permissions .....	1
Service quotas .....	2
<b>Logging .....</b>	<b>3</b>
Logging with CloudTrail .....	3
Agreements API information in CloudTrail .....	3
Understanding Agreements API actions .....	4
Understanding Agreements API log file entries .....	5
<b>Actions .....</b>	<b>11</b>
DescribeAgreement .....	12
Request Syntax .....	12
Request Parameters .....	12
Response Syntax .....	12
Response Elements .....	13
Errors .....	15
Examples .....	16
See Also .....	17
GetAgreementTerms .....	18
Request Syntax .....	18
Request Parameters .....	18
Response Syntax .....	19
Response Elements .....	19
Errors .....	20
Examples .....	21
See Also .....	23
SearchAgreements .....	24
Request Syntax .....	24
Request Parameters .....	25
Response Syntax .....	27
Response Elements .....	27
Errors .....	28
Examples .....	28
See Also .....	30

<b>Data Types</b> .....	<b>31</b>
AcceptedTerm .....	33
Contents .....	33
See Also .....	35
Acceptor .....	36
Contents .....	36
See Also .....	36
AgreementViewSummary .....	37
Contents .....	37
See Also .....	39
ByolPricingTerm .....	40
Contents .....	40
See Also .....	40
ConfigurableUpfrontPricingTerm .....	41
Contents .....	41
See Also .....	42
ConfigurableUpfrontPricingTermConfiguration .....	43
Contents .....	43
See Also .....	43
ConfigurableUpfrontRateCardItem .....	44
Contents .....	44
See Also .....	44
Constraints .....	46
Contents .....	46
See Also .....	46
Dimension .....	47
Contents .....	47
See Also .....	47
DocumentItem .....	49
Contents .....	49
See Also .....	50
EstimatedCharges .....	51
Contents .....	51
See Also .....	52
Filter .....	53
Contents .....	53

See Also .....	53
FixedUpfrontPricingTerm .....	55
Contents .....	55
See Also .....	56
FreeTrialPricingTerm .....	57
Contents .....	57
See Also .....	57
GrantItem .....	59
Contents .....	59
See Also .....	59
LegalTerm .....	60
Contents .....	60
See Also .....	60
PaymentScheduleTerm .....	61
Contents .....	61
See Also .....	62
ProposalSummary .....	63
Contents .....	63
See Also .....	63
Proposer .....	64
Contents .....	64
See Also .....	64
RateCardItem .....	65
Contents .....	65
See Also .....	65
RecurringPaymentTerm .....	66
Contents .....	66
See Also .....	67
RenewalTerm .....	68
Contents .....	68
See Also .....	68
RenewalTermConfiguration .....	69
Contents .....	69
See Also .....	69
Resource .....	70
Contents .....	70

---

See Also .....	70
ScheduleItem .....	72
Contents .....	72
See Also .....	72
Selector .....	73
Contents .....	73
See Also .....	73
Sort .....	74
Contents .....	74
See Also .....	74
SupportTerm .....	75
Contents .....	75
See Also .....	75
UsageBasedPricingTerm .....	76
Contents .....	76
See Also .....	76
UsageBasedRateCardItem .....	78
Contents .....	78
See Also .....	78
ValidationExceptionField .....	79
Contents .....	79
See Also .....	79
ValidityTerm .....	80
Contents .....	80
See Also .....	81
<b>Common Parameters .....</b>	<b>82</b>
<b>Common Errors .....</b>	<b>85</b>
<b>Document history .....</b>	<b>87</b>

# AWS Marketplace Agreement Service

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.

## Topics

- [Supported AWS Regions](#)
- [Permissions](#)
- [Service quotas](#)

## Supported AWS Regions

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
```

## Permissions

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*.

## Service quotas

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



# Logging

The AWS Marketplace Agreement Service supports logging with AWS CloudTrail.

## Topics

- [Logging Agreements API calls using AWS CloudTrail](#)

## Logging Agreements API calls using AWS 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` 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": "ABCDEFGHijklmnop123"
  },
  "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": "ABCDEFGHijklmnop12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHijklmnop1234"
  },
  "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": "ABCDEFGHijklmnop12345",
    "arn": "arn:aws:iam::123456789010:user/CloudtrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHijklmnop123",
  },
  "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-6wy4rhp7l6iyuu2jrcgd1shdi"
  },
  "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": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudtrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP123",
  },
}
```

```
  },
  "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",  
}
```



# Actions

The following actions are supported:

- [DescribeAgreement](#)
- [GetAgreementTerms](#)
- [SearchAgreements](#)

# DescribeAgreement

Provides details about an agreement, such as the proposer, acceptor, start date, and end date.

## Request Syntax

```
{
  "agreementId": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

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

## 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.
- ROLLED\_BACK (Only applicable to inactive agreement revisions) – The agreement revision has been rolled back because of an error. An earlier revision is now active.
- SUPERCEDED (Only applicable to inactive agreement revisions) – The agreement revision is no longer active and another agreement revision is now active.
- 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 | ROLLED\_BACK | SUPERSEDED | 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

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

For information about the parameters that are common to all actions, see [Common Parameters](#).

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

### 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

### **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 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

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

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### 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.
- `AcceptorAccountId` – 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

### **InternalServerError**

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": "ABCDEFGHIJKLMN0P123"
    },
    "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](#)

# Data Types

The AWS Marketplace Agreement Service API contains several data types that various actions use. This section describes each data type in detail.

## Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [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

A subset of terms proposed by the proposer, which have been accepted by the acceptor as part of agreement creation.

## Contents

### 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

The details of the party accepting the agreement terms. This is commonly the buyer for PurchaseAgreement.

## Contents

### 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

A summary of the agreement, including top-level attributes (for example, the agreement ID, version, proposer, and acceptor).

## Contents

### 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

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

### 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

Defines a prepaid payment model that allows buyers to configure the entitlements they want to purchase and the duration.

## Contents

### 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

Defines a prepaid payment model that allows buyers to configure the entitlements they want to purchase and the duration.

## Contents

### 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

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

### 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

Defines limits on how the term can be configured by acceptors.

## Contents

### **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

Defines the dimensions that the acceptor has purchased from the overall set of dimensions presented in the rate card.

## Contents

### **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

Includes the list of references to legal resources proposed by the proposer to the acceptor. Each DocumentItem refers to an individual reference.

### Contents

#### 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

Estimated cost of the agreement.

## Contents

### 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

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

### 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

Defines a prepaid pricing model where the customers are charged a fixed upfront amount.

## Contents

### 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

Defines a short-term free pricing model where the buyers aren't charged anything within a specified limit.

## Contents

### 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

Entitlements granted to the acceptor of fixed upfront as part of agreement execution.

## Contents

### 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

Defines the list of text agreements proposed to the acceptors. An example is the end user license agreement (EULA).

### Contents

#### 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

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

### 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

A summary of the proposal received from the proposer.

## Contents

### 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

Details of the party proposing the agreement terms,. This is commonly the seller for PurchaseAgreement.

## Contents

### 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

Defines the per unit rates for each individual product dimension.

## Contents

### 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

Defines a pricing model where customers are charged a fixed recurring price at the end of each billing period.

## Contents

### 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

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

### 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

Additional parameters specified by the acceptor while accepting the term.

## Contents

### **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

The list of resources involved in the agreement.

## Contents

### 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

An individual installment of the payment that includes the date and amount of the charge.

## Contents

### 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

Differentiates between the mutually exclusive rate cards in the same pricing term to be selected by the buyer.

## Contents

### 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

An object that contains the `SortBy` and `SortOrder` attributes.

## Contents

### `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

Defines the customer support available for the acceptors when they purchase the software.

## Contents

### 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

Defines a usage-based pricing model (typically, pay-as-you-go pricing), where the customers are charged based on product usage.

## Contents

### 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

Within the pay-as-you-go model defined under `UsageBasedPricingTerm`, the `UsageBasedRateCardItem` defines an individual rate for a product dimension.

## Contents

### 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

The input fails to satisfy the constraints specified by the service.

## Contents

### 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

Defines the conditions that will keep an agreement created from this offer valid.

## Contents

### **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](#)

# 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
<a href="#">The AWS Marketplace Agreement Service API reference is now generally available</a>	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