

Unable to locate subtitle

AWS Marketplace Deployment Service



AWS Marketplace Deployment Service: *Unable to locate subtitle*****

Table of Contents

AWS Marketplace Deployment Service	1
Supported AWS Regions	1
Permissions	1
Service quotas	2
Logging	4
Logging with AWS CloudTrail	4
AWS Marketplace Deployment Service information in CloudTrail	4
Understanding AWS Marketplace log file entries	5
Actions	7
ListTagsForResource	8
Request Syntax	8
URI Request Parameters	8
Request Body	8
Response Syntax	8
Response Elements	8
Errors	9
See Also	9
PutDeploymentParameter	11
Request Syntax	11
URI Request Parameters	11
Request Body	12
Response Syntax	13
Response Elements	14
Errors	15
See Also	16
TagResource	17
Request Syntax	17
URI Request Parameters	17
Request Body	17
Response Syntax	17
Response Elements	18
Errors	18
See Also	19
UntagResource	20

Request Syntax	20
URI Request Parameters	20
Request Body	20
Response Syntax	20
Response Elements	20
Errors	20
See Also	21
Data Types	23
DeploymentParameterInput	24
Contents	24
See Also	24
Common Parameters	26
Common Errors	29
Document history	31

AWS Marketplace Deployment Service

The AWS Marketplace Deployment Service supports the Quick Launch experience, which is a deployment option for software as a service (SaaS) products. Quick Launch simplifies and reduces the time, resources, and steps required to 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 during the Quick Launch experience.

Note

For information about the Quick Launch experience for sellers, see [Configure Quick Launch](#) in the *AWS Marketplace Seller Guide*.

Topics

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

Supported AWS Regions

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

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](#).

Service quotas

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

Quota	Default	Description
		deployment parameter per 24 hours
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

The AWS Marketplace Deployment Service supports logging with AWS CloudTrail.

Topics

- [Logging AWS Marketplace Deployment Service calls with CloudTrail](#)

Logging AWS Marketplace Deployment Service 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": "ABCDEFGHijklmnop12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHijklmnop123"
  },
  "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"
}
```

Actions

The following actions are supported:

- [ListTagsForResource](#)
- [PutDeploymentParameter](#)
- [TagResource](#)
- [UntagResource](#)

ListTagsForResource

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

InternalServerError

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

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

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

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

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

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

InternalServerError

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

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](#)

Data Types

The AWS Marketplace Deployment 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:

- [DeploymentParameterInput](#)

DeploymentParameterInput

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

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](#)

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
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