Message Queue (MQ)

O&M APIs

O&M APIs

Release notes

API version: 2.0.1

- Added the MQTT GroupId creating interface.

API version: 2.0.0

- Added the Message Tracing interface.
- Added the access permission control to the write operation interface.

API version: 1.3.3

- Added the Topic read-/write-prohibition interface.
- Added the ConsumerId reading-prohibition interface.

API version: 1.3.2-SNAPSHOT

- Modified the maximum API interface timeout time to 10s.
- Added the alarm editing interface.

API version: 1.3.1-SNAPSHOT

- Added support to MQTT-related O&M API.
- Added the user traffic limiting function to all APIs.

API version: 1.2.1-SNAPSHOT

- Allows to view the authorization interface without specifying the UserId.
- Supported querying the ParentId of the called account.

API version: 1.2.0-SNAPSHOT

- Supported region-based calling.
- Supported the account to authorize APIs.
- Supported the query API regarding message sending/receiving trend.

API version: 1.1.1-SNAPSHOT

- Supported querying messages by pages according to the Topic.
- Supported clearing the accumulated messages in two ways: one is to clear all messages at once, and the other is to clear them according to the timestamps.

API version: 1.1.0-SNAPSHOT

- Open API supported dynamic error codes.
- TopicGet interface no longer required Region, and can be used to query the Region where a certain Topic belongs.
- The version is upgraded to an HTTPS interface. Version lower than 1.1.0-SNAPSHOT is no longer supported.
- Changed the version to 1.1.0-SNAPSHOT.

API version: 1.0.2-SNAPSHOT

- Fixed a few bugs of Open API:
    - Exception information passes through when a duplicate Topic is created.
    - A Topic needs to be transferred in when querying a message.
- Added information required for authentication.
- Added an API interface for Message Tracing.

API version: 1.0.1-SNAPSHOT

- Fixed a few bugs of Open API:
    - Replaced all regionIds in the return result with OnsRegionId to distinguish them from the regionIds of POP to help users tell the difference.
    - Fixed several issues of the consumerstatus return result, and other issues such as protecting consumetime out of bounds and no description for clientId.
    - Fixed issues such as consumerstatus and other APIs returning empty lists.
- Changed the version to 1.0.1-SNAPSHOT.

API version: 1.0.0-SNAPSHOT

- Open API online test.
- Updated parameter descriptions in the documentation and fixed the use of methods such as regionidList.
- Added the dependency description of accessing API.

Access instruction

This topic explains how to access MQ Open API, including how to obtain SDK and set initialization parameters.

1. Obtain SDK

Use the following POM configuration to add dependency on the SDK of Open API.

```
<dependencies>
<dependency>
<groupId>com.aliyun</groupId>
<artifactId>aliyun-java-sdk-core</artifactId>
<optional>true</optional>
<version>3.2.8</version>
</dependency>
<dependency>
<groupId>com.aliyun</groupId>
<artifactId>aliyun-java-sdk-ons</artifactId>
<version>2.0.1</version>
</dependency>
</dependencies>
```

2. Public parameter setting

To use Open API, start the Open API client, and when you do this, set the access point and parameters such as AccessKey and SecretKey. Here’ s an example:

```
    /**
*Open API's access point; set as the target Region
*/
String regionId = "XXXXX";
/**
*AccessKey used for authenticating; obtained from the Console on the Alibaba Cloud official website
*/
String accessKey = "XXXXXXXXXXXXXXXXX";
/**
*SecretKey used for authenticating; obtained from the Console on the Alibaba Cloud official website
*/
String secretKey = "XXXXXXXXXXXXXXXXX";
/*
*The name of cloud products accessed through Open API; set as Ons
*/
String productName ="Ons";
/**
*The access point domain name corresponding to the endPoint access point
*/
String domain ="ons.XXXXX.aliyuncs.com";
try {
DefaultProfile.addEndpoint(regionId , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
```

Parameter descriptions

- regionId: the region where the API gateway is located, currently supported ones are cn-beijing, cn-hangzhou, cn-qingdao, cn-shenzhen, etc.
- accessKey: AK obtained by a user on the Alibaba ECS Management Console.
- secretKey: SK obtained by a user on the Alibaba ECS Management Console.
- endPointName: the access point name, consistent with RegionId.
- productName: the product name of Open API. Can be simply set as Ons.
- domain: the access point domain of Open API. The rule is ons.${RegionId}.aliyuncs.com. Regions with Open API support are summarized in the following table.

Region Name	RegionId	Domain	Note
North China 2 in public cloud	cn-beijing	ons.cn-beijing.aliyuncs.com	Recommended for users who use the Region of North China 2 in public cloud.
Internet	cn-qingdao	ons.cn-qingdao.aliyuncs.com	Recommended for users who use the Region of Internet.
East China 1 in public cloud	cn-hangzhou	ons.cn-hangzhou.aliyuncs.com	Recommended for users who use the Region of East China 1 in public cloud.
East China 2 in public cloud	cn-shanghai	ons.cn-shanghai.aliyuncs.com	Recommended for users who use the Region of East China 2 in public cloud.
South China 1 in public cloud	cn-shenzhen	ons.cn-shenzhen.aliyuncs.com	Recommended for users who use the Region of South China 1 in public cloud.
Singapore in public cloud	ap-southeast-1	ons.ap-southeast-1.aliyuncs.com	Recommended for users who use the Region of Singapore in public cloud.
East China 1 in AntCloud	cn-hangzhou-finance	ons.cn-hangzhou-finance.aliyuncs.com	Recommended for users who use the Region of East China 1 in AntCloud.
South China 1 in AntCloud	cn-shenzhen-finance	ons.cn-shenzhen-finance.aliyuncs.com	Recommended for users who use the Region of South China 1 in AntCloud.
East China 2 in AntCloud	cn-shanghai-finance	ons.cn-shanghai-finance.aliyuncs.com	Recommended for users who use the Region of East China 2 in AntCloud.

Terms

This topic explains common terms involved in MQ Open API for reference.

Gateway parameters

RegionId

It refers to the region where Alibaba Cloud products are located, including cn-beijing, cn-hangzhou, etc. You need to set the corresponding region before calling Open API. For example, you need to set the region of cn-beijing before operating the MQ resource in north China 2. Refer to the table in Homepage for related information.

OnsRegionId

It refers to the information of the region defined by MQ products. It is used for parameters requested by API, and specifying API to operate MQ resources in corresponding regions. The information of the region needs to be obtained according to the OnsRegionList interface.

EndPoint

Open API calls the corresponding back-end service according to the set access point. It is set as the corresponding RegionId.

Domain

Open API calls the corresponding backend service according to the set access domain name. For example, the domain is “ ons.cn-hangzhou.aliyuncs.com”, which indicates that the MQ backend service of east China 1 is linked. Refer to Access Guide for specific domain name information. (~~44419~~)

Account security parameters

AccessKey

AK of the account used for accessing Alibaba Cloud products. It is obtained from the Alibaba Cloud official website. It is transferred to the MQ backend for identity authentication during calls. Only resources in corresponding accounts can be operated when calling MQ Open API.

SecretKey

It refers to an AK secret key of the account used for accessing Alibaba Cloud products. It is obtained from the Alibaba Cloud official website and transferred during calls for calculating the signature of the access request. MQ backend compares the corresponding account with the signature for authentication.

MQ resource management and control parameters

Topic

Messages in MQ products are stored by Topic. Therefore, Topic is one of the most important resources in MQ. Topic is globally unique in MQ.

ProducerId (PID)

It is the unique ID for determining the publish relation in MQ products, and is used to identify a sent cluster. ProducerId is globally unique in MQ. PID is how it is referenced in subsequent documents.

ConsumerId (CID)

It is the unique ID for determining the subscribe relation in MQ products, and is used to identify a consumer. Logic of consumed messages of clients using the same ConsumerId must be completely consistent. ConsumerId is globally unique in MQ. CID is how it is referenced in subsequent documents.

Message Accumulation

It describes message data in MQ products which has not been processed by consumers. When message production is faster than message consumption, the difference between the produced message and the consumed message is the number of accumulations. Under normal conditions, the number of accumulations should be within a controllable range. If the number of accumulations is too large and seems to increase gradually, you need to focus on the reason why consumers consume messages slowly.

Scenarios

Resource query

This topic explains how to use an Open API to query MQ resources (including Topic, Publish, and Subscribe) owned by a specified account.

Resource query

The queried resources include Topic, Publish, and Subscribe information.

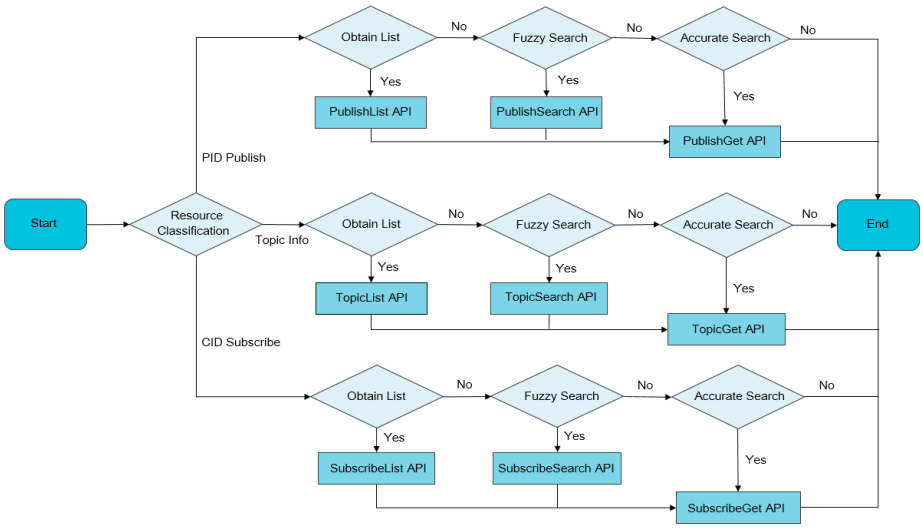
You can perform an exact match query with the name of Topic, Publish, and Subscribe.

If no precise information is available, then automatically search for and query according to the related content.

If precise information is not required, you can query the list of all resources under the current account instead.

Main process

The calling process is shown in the following figure:



Related APIs

- Publish list: OnsPublishList
- Publish search: OnsPublishSearch
- Publish exact match query: OnsPublishGet
- Topic list: OnsTopicList
- Topic search: OnsTopicSearch
- Topic exact match query: OnsTopicGet
- Subscribe list: OnsSubscribeList
- Subscribe search: OnsSubscribeSearch
- Subscribe exact match query: OnsSubscribeGet

Message accumulation query

This topic explains how to use Open API to query the MQ message status in the running process, including the message sending status, message consumption status, the presence of accumulated messages, how to handle accumulation, and so on.

Query method

To view the Topic message sending status: view the total number of valid messages, and the latest status of the messages of the target Topic on the current server.

To view the consumption progress of the Consumer ID: view if the consumption of the specified Consumer ID is accumulated.

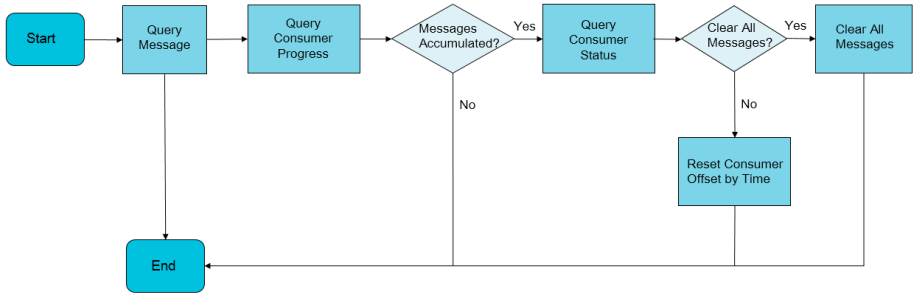
Client consumption status query: if the client consumption is accumulated, view the connection status of clients in the consumption group, and determine if consumer threads of the client are blocked.

Consumer offset resetting: update the consumption progress offset of the consumer according to business requirements. This function includes two scenarios.

- Scenario A: clear all messages if they can be discarded.
- Scenario B: reset the offset according to the specified time if the accumulated messages need to be recycled or updated to a specified time.

Main process

The calling process is shown in the following figure:



Related APIs

- Current message query: OnsTopicStatus
- Consumption progress query: OnsConsumerAccumulate

- Consumer status query: OnsConsumerStatus
- Clear accumulation: OnsConsumerResetOffset

Resource authorization

This topic explains how to perform authorization-related tasks for Topic resource with Open API.

Resource authorization

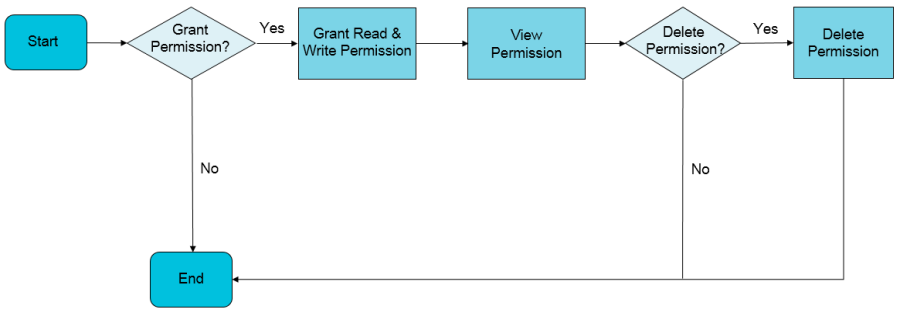
Create authorization: grants the target user, a primary account or a sub-account, the read and/or write permission of your currently owned Topic.

View authorization: views the list of users who have been granted the permission of the target Topic, and the permission data.

Delete authorization: deletes authorization of a certain Topic on the target account.

Main process

The calling process is shown in the following figure:



Related APIs

- Create authorization: OnsEmpowerCreate
- View authorization: OnsEmpowerList
- Delete authorization: OnsEmpowerDelete

Error codes

A successful calling of MQ Open API returns ResponseCode=200 to the client, and a failed one returns an error code and a description. You can find solutions accordingly in the table below.

Error Code	Symptom and Cause	Solution
ONS_SYSTEM_ERROR	MQ backend exception	Contact MQ technical support through an Alibaba Cloud ticket
ONS_SERVICE_UNSUPPORTED	The calling is not supported in the corresponding Region	Check the interface activation status with MQ technical support
ONS_INVOKE_ERROR	Open API interface calling failed	Contact MQ technical support
BIZ_FIELD_CHECK_INVALID	Parameter verification failed	Refer to API manual to check if each input parameters is valid
BIZ_TOPIC_NOT_FOUND	Topic is not found	Check if the Topic input is valid or if it was created
BIZ_SUBSCRIPTION_NOT_FOUND	The CID of the target subscription relation is not found	Check if the CID was created, or if the query criterion is incorrect
BIZ_PUBLISHER_EXISTED	The specified PID already exists	Change the PID name and retry the request
BIZ_SUBSCRIPTION_EXISTED	The specified CID already exists	Change the CID name and retry the request
BIZ_CONSUMER_NOT_ONLINE	The client of the specified CID is offline	Make sure the consumer client is online and retry the request
BIZ_NO_MESSAGE	No matching message is found for the current query criterion	Check the query criterion and confirm if a message was sent within the query range
BIZ_REGION_NOT_FOUND	The requested Region is not found	Confirm if the requested Region parameter is valid
BIZ_TOPIC_EXISTED	The specified Topic already exists	Change the Topic name and retry the request
BIZ_PRODUCER_ID_BELONG_TO_OTHER_USER	The current PID has been taken by other users	Change the PID and retry the request
BIZ_CONSUMER_ID_BELONG_TO_OTHER_USER	The current CID has been taken by other users	Change the CID and retry the request

BIZ_PUBLISH_INFO_NOT_FOUND	The requested PID is not found	Confirm if the PID exists or if the request criterion is incorrect
EMPOWER_EXIST_ERROR	The current authorization relation already exists	Check the requested parameter and retry, or query the requested parameter first
EMPOWER_OWNER_CHECK_ERROR	The current user is not the owner of the authorized Topic	Confirm the resource ownership

User management APIs

Fetch Region Information

OnsRegionList interface returns a list of currently available Regions of MQ.

Application scenario

Region information is required when calling all Open APIs. The Region field can be retrieved through the OnsRegionList interface.

List of request parameters

Name	Type	Required	Description
OnsRegionId	String	No	The region of the currently queried MQ.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.

List of response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(RegionDo)	The query result

RegionDo data structure

Member	Type	Description
Id	Long	Number in a database
RegionId	String	The name of RegionId
RegionName	String	The alias of Region
ChannelId	Long	The number of the channel
ChannelName	String	The alias of the channel
CreateTime	Long	Time of creation
UpdateTime	Long	Time of the last update

Example

```
public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.

*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient= new DefaultAcsClient(profile);
OnsRegionListRequest request = new OnsRegionListRequest();
```

```
request.setAcceptFormat(FormatType.JSON);
request.setPreventCache(System.currentTimeMillis());
try {
OnsRegionListResponse response = IAcsClient.getAcsResponse(request);
List<OnsRegionListResponse.RegionDo> regionDoList=response.getData();
for (OnsRegionListResponse.RegionDo regionDo:regionDoList){
System.out.println(regionDo.getId()+" "+
regionDo.getOnsRegionId()+" "+
regionDo.getRegionName()+" "+
regionDo.getChannelId()+" "+
regionDo.getChannelName()+" "+
regionDo.getCreateTime()+" "+
regionDo.getUpdateTime());
}
} catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Topic management APIs

Create Topic

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

Topic creation interface is usually used when new Topics are required. For example, to publish a new application, new Topics are required for the business logic.

Topic is globally unique and must not be duplicate; otherwise, the creation fails, where you need to fix the issue according to the error code.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts by default, and cannot be used by RAM

sub-accounts unless they are granted related permission.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. For more information, see Terms.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	The name of the Topic to be created, which is globally unique and must not be duplicate.
Qps	Long	No	Estimated QPS of the Topic.
Status	Integer	No	The creation status of a Topic (0 for “in service”, 1 for “frozen”, and 2 for “paused”.)
Remark	String	No	A optional note
Appkey	String	No	The key of an application (optional for public cloud users, and required for Tmall Cloud users).
AppName	String	No	An application name (optional for public cloud users).

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request, used for

		troubleshooting and identifying issues.
HelpUrl	String	A help link

Related APIs

- OnsTopicDelete: deletes a Topic
- OnsTopicList: views the list of Topics
- OnsPublishCreate: creates a publishing relation after creating a Topic
- OnsSubscribeCreate: creates a subscription after creating a Topic

Example

As a reference, this example shows the process of accessing from the end point *China East*, and creating a Topic with the name of “XXX” in the *daily* region.

```
    public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);

OnsTopicCreateRequest request = new OnsTopicCreateRequest();
request.setAcceptFormat(FormatType.JSON);
request.setTopic("XXXXXXXX");
request.setQps(1000l);
request.setRemark("test");
request.setStatus(0);
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
try {
```

```
OnsTopicCreateResponse response = client.getAcsResponse(request);
System.out.println(response.getRequestId());
}
catch (ServerException e) {
e.printStackTrace();
}
catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Delete Topic

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsTopicDelete interface deletes a specified Topic in the account.

This interface is used for recycling resources when, for example, the application is offline. When deleting a Topic, the MQ backend slows down due to recycling resources, so it’ s not recommended to re-create the Topic immediately after deleting it. If Topic deletion fails, please process it according to the error code.

NOTE: deleting a Topic also removes all of its connections with Producer IDs/Consumer IDs. Call this API with caution.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts by default, and cannot be used by RAM sub-accounts unless they are granted related permission.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. It can be

			obtained with the OnsRegionList method. For more information, see Terms.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Specify the Topic to be deleted.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request, used for troubleshooting and identifying issues.
HelpUrl	String	A help link

Related APIs

- OnsTopicCreate: creates a Topic
- OnsTopicList: views the list of Topics
- OnsPublishDelete: deletes a publishing relation
- OnsSubscribeDelete: deletes a subscription

Example

As a reference, this example shows the process of accessing from the end point *China East*, and deleting a Topic with the name of “Mingduan_67dd”.

```
   public static void main(String[]args){
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";
```

```
/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTopicDeleteRequest request =new OnsTopicDeleteRequest();
request.setCluster("taobaodaily");
request.setPreventCache(System.currentTimeMillis());
/**
*OnsRegionId refers to the particular region of MQ set to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setTopic("Mingduan_67dd");
try {
OnsTopicDeleteResponse response = client.getAcsResponse(request);
System.out.println(response.getRequestId());
}
catch (ServerException e) {
e.printStackTrace();
}
catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Fetch Topic list

OnsTopicList interface returns a list of all Topics of the current user.

Scenario

This interface returns a list of all Topics in the account, and is usually used for generating a list of resources instead of viewing detailed information.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. It can be obtained with the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
Topic	String	No	Specify this to query a specific Topic, or don’t specify to query all Topics.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	Returns a list of all published Topics

PublishInfoDo data structure

Member	Type	Description
id	Long	The number of the Topic
channelId	Integer	The ID of the region where the Topic is located. 0 for Alibaba Cloud, and 1 for Tmall Cloud.
channelName	String	The ID of the region where the Topic is located. ALIYUN for Alibaba Cloud, and CLOUD for Tmall Cloud ...
onsRegionId	Long	The ID of the region where

		the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner; Alibaba Cloud’s uid
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for “publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status (0 for “in service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status (“in service”, “frozen”, and “paused”)
appkey	String	null
createTime	Long	Creation time
updateTime	Long	Update time
remark	String	Note

Related APIs

- OnsTopicCreate: creates a Topic
- OnsTopicList: views the list of Topics
- OnsPublishDelete: deletes a publish relation
- OnsSubscribeDelete: deletes a subscription relation

Example

As a reference, this example shows the process of accessing from the end point *China East*, and querying the list of all Topics of the current user.

```
public static void main (String[]args) {
```

```
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile = DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient = new DefaultAcsClient(profile);
OnsTopicListRequest request = new OnsTopicListRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setTopic("XXXXXXXXXXXXX");
try {
OnsTopicListResponse response = IAcsClient.getAcsResponse(request);
List<OnsTopicListResponse.PublishInfoDo> publishInfoDoList=response.getData();
for(OnsTopicListResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getTopic()+" "+ publishInfoDo.getOwner());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query Topic status

OnsTopicGet interface queries the information of a specified Topic. It’ s an exact match query.

Usage

This interface is used for querying the location of a specified Topic and obtaining information such as its activation status and permission relations. Because it’ s an exact match query, the Topic name is

required in the input parameter.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. It can be obtained with the OnsRegionList method. For more information, see Terms.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Specify the Topic to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	Returns a list of all published Topics

PublishInfoDo data structure

Member	Type	Description
id	Long	Information No. of the Topic
channelId	Integer	The ID of the region where the Topic is located. 0 for Alibaba Cloud, and 1 for Tmall Cloud.
channelName	String	The category of the Topic, which can be Alibaba Cloud or Tmall Cloud.

onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner; Alibaba Cloud’s UID
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for “publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status (0 for “in service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status (“in service”, “frozen”, and “paused”)
appkey	String	null
createTime	Long	Creation time
updateTime	Long	Update time
remark	String	Note

Related APIs

- OnsTopicCreate: creates a Topic
- OnsTopicList: views the list of Topics
- OnsPublishDelete: deletes a publishing relation
- OnsSubscribeDelete: deletes a subscription relation

Example

As a reference, this example shows the process of accessing from the end point *China East*, querying the Topic “MingduanTest”, and printing the Topic-related information.

```
    public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient= new DefaultAcsClient(profile);

OnsTopicGetRequest request = new OnsTopicGetRequest();
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
try {
OnsTopicGetResponse response = IAcsClient.getAcsResponse(request);
List<OnsTopicGetResponse.PublishInfoDo> publishInfoDoList=response.getData();
for (OnsTopicGetResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getId()+" "+
publishInfoDo.getChannelId()+" "+
publishInfoDo.getChannelName()+" "+
publishInfoDo.getOnsRegionId()+" "+
publishInfoDo.getRegionName()+" "+
publishInfoDo.getTopic()+" "+
publishInfoDo.getOwner()+" "+
publishInfoDo.getRelation()+" "+
publishInfoDo.getRelationName()+" "+
publishInfoDo.getStatus()+" "+
publishInfoDo.getStatusName()+" "+
publishInfoDo.getAppkey()+" "+
publishInfoDo.getCreateTime()+" "+
publishInfoDo.getUpdateTime()+" "+
publishInfoDo.getRemark());
}
System.out.println(response.getRequestId());
}
catch (ServerException e) {
e.printStackTrace();
}
catch (ClientException e) {
e.printStackTrace();
```

```
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Query Topic details

OnsTopicSeacrh interface returns a list of all Topics in the specified account according to the search criteria.

Usage

This interface is used for querying Topics without knowing Topic names, and returns a list of all Topics. An exact match query can then be performed with the OnsTopicGet interface for a specified Topic.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. It can be obtained with the OnsRegionList method. For more information, see Terms.
OnsPlatform	String	No	The request sources from the POP platform by default.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Search	String	Yes	Input search criteria such as Topic field or appname field.

Response parameters

Name	Type	Description

RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	Returns a list of all published Topics

PublishInfoDo data structure

Member	Type	Description
id	Long	Information No. of the Topic
channelId	Integer	The ID of the region where the Topic is located. 0 for Alibaba Cloud, and 1 for Tmall Cloud.
channelName	String	The name of the region where the Topic is located, including ALIYUN, CLOUD...
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner; Alibaba Cloud’ s UID
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for “publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status (0 for “in Service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status(“in service”, “frozen”, and “paused”
appkey	Integer	null
createTime	Long	Creation time
updateTime	Long	Update time

remark	String	Note information

Related APIs

- OnsTopicCreate: creates a Topic
- OnsTopicList: views the list of Topics
- OnsPublishDelete: deletes a publishing relation
- OnsSubscribeDelete: deletes a subscription relation

Example

As a reference, this example shows the process of accessing from the end point *China East*, and querying all Topics of the current user.

```
   public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient= new DefaultAcsClient(profile);

OnsTopicSearchRequest request = new OnsTopicSearchRequest();
request.setAcceptFormat(FormatType.JSON);
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setSearch("JODIE_TEST_28dd");
try {
OnsTopicSearchResponse response = IAcsClient.getAcsResponse(request);
List<OnsTopicSearchResponse.PublishInfoDo> publishInfoDoList=response.getData();
for(OnsTopicSearchResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getTopic()+" "+publishInfoDo.getOwner());
}
System.out.println(response.getRequestId());
```

```
} catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Query Topic's consumer offset

This topic explains the OnsTopicStatus interface, which queries the current message status of a specified Topic.

Usage

This interface queries the total number of messages in the current Topic, and the last update time. It can gauge the usage rate of Topic resources. The TopicStatus interface returns the number of all messages in the specified Topic on the current server, and the last message writing time.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	Set the region to be queried; its value is obtained through the OnsRegionList interface
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	The name of the queried Topic.

Response parameters

Name	Type	Description

RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	OnsTopicStatusResponse.Data	The information storage data structure of the queried Topic.

OnsTopicStatusResponse.Data data structure

Member	Type	Description
totalCount	Long	The total number of the messages in all partitions of the current Topic.
lastTimeStamp	Long	The last update time of the current Topic.

Related APIs

- OnsTopicCreate: creates a Topic
- OnsConsumerStatus: queries the consumer status of the Consumer ID

Example

As a reference, this example shows the process of accessing from the end point *China East*, and querying the number of messages in the Topic “JODIE_TEST_27dd” of the current user.

```
public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTopicStatusRequest request = new OnsTopicStatusRequest();
request.setAcceptFormat(FormatType.JSON);
```

```
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setTopic("JODIE_TEST_27dd");
try {
OnsTopicStatusResponse response = iAcsClient.getAcsResponse(request);
OnsTopicStatusResponse.Data data=response.getData();
Long totalCount =data.getTotalCount();
Long lastTimeStamp =data.getLastTimeStamp();
System.out.println(response.getRequestId());
} catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Configure Topic permission

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Usage

This interface is used for configuring read/write switches of the Topic based on the specified MQ region and the Topic name. It’ s usually used for write-only or read-only scenarios.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts by default, and cannot be used by RAM sub-accounts unless they are granted related permission.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It

			can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	The name of the Topic required to be configured, which must not be duplicate under the same user even in different regions.
Perm	Integer	Yes	Controls the read/write permission of the Topic. 6 for “read/write access”, 4 for “read-only”, and 2 for “write-only”.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsTopicCreate: creates a Topic
- OnsTopicList: views the list of Topics

Example

As a reference, this example shows the process of accessing from the end point *China East*, and setting the Topic “MinqduanTest” to write-only mode.

```
    public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient= new DefaultAcsClient(profile);

OnsTopicUpdateRequest request = new OnsTopicUpdateRequest();
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setPerm(4);//2 indicates "write-only"; 4 indicates "read-only"; 6 indicates "read/write supported".
try {
OnsTopicUpdateResponse response = IAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
}
catch (ServerException e) {
e.printStackTrace();
}
catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Production management APIs

Create PID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsPublishCreate interface creates and registers a publishing relation to MQ, and obtain the generated PID for sending messages in the production environment.

Usage

After creating a Topic, this interface is called when the corresponding PID needs to be registered for newly created resources to send messages.

Notes: each Topic only needs one PID. There’ s no need to create multiple PIDs because a PID can be reused in multiple applications.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region of the currently queried MQ. It can be obtained with the OnsRegionList method. For more information, see Terms.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ProducerId	String	Yes	PID of the publishing relation to be created.
Topic	String	Yes	The name of the published Topic.

Return parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsPublishDelete: deletes a publishing relation
- OnsPublishList: retrieves a list of publishing relations
- OnsPublishGet: queries a publishing relation

Example

This example shows the process of accessing from the end point *China East*, and creating a publishing relation named “PID_Mingduan” in the public beta region. This publishing relation is used for sending messages of Topic “MingduanTestAPI”.

```
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsPublishCreateRequest request = new OnsPublishCreateRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("publictest");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTestAPI");
```

```
request.setProducerId("PID_Mingduan");
try {
OnsPublishCreateResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Return result

CC815D34-0950-43BF-A4E8-4187C3C7BF19

Delete PID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsPublishDelete interface deletes published PIDs.

Usage

This interface is called to delete the existing publishing relation when an application is disabled or the resource needs to be released. Make sure that the publishing relation already exists before deleting.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by

			default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ProducerId	String	No	PID of the publishing relation to be deleted.
Topic	String	Yes	Topic corresponding to the publishing relation to be deleted.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsPublishCreate: creates a publishing relation
- OnsPublishList: retrieves a list of publishing relations
- OnsPublishGet: queries a publishing relation

Example

The following example is to delete a relation of a PID with the name of “PID_MingduanTest” in the daily region.

This example shows the process of deleting a PID relation named “PID_MingduanTest” in the daily region.

```
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
```

```
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsPublishDeleteRequest request = new OnsPublishDeleteRequest();
/**
*OnsRegionId refers to the resource of the region of ONS required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("Mingduan_67dd");
request.setProducerId("PID_MingduanTest");
try {
OnsPublishDeleteResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Fetch all PIDs of specified user

OnsPublishList interface retrieves all publishing relations owned by a specified user.

Usage

This interface shows a list of publishing relations owned by a user.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList

			interface.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	A set of query results

PublishInfoDo data structure

Member	Type	Description
id	Long	ID of the publishing information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	The name of the region where the Topic is located. ALIYUN for Alibaba Cloud, CLOUD for Tmall Cloud...
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for

		“publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status. (0 for “in service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status (“in service”, “frozen”, and “paused”)
appkey	Integer	null
createTime	Long	Creation time
updateTime	Long	Update time
remark	String	An optional note

Related APIs

- OnsPublishDelete: deletes a publishing relation
- OnsPublishGet: queries a publishing relation
- OnsPublishCreate: creates a publishing relation

Example

This example shows the process of querying and printing a list of all PIDs of a user in the daily region.

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
```

```
OnsPublishListRequest request = new OnsPublishListRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
try {
OnsPublishListResponse response=iAcsClient.getAcsResponse(request);
List<OnsPublishListResponse.PublishInfoDo> publishInfoDoList =response.getData();
for (OnsPublishListResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getId()+" "+
publishInfoDo.getChannelId()+" "+
publishInfoDo.getChannelName()+" "+
publishInfoDo.getOnsRegionId()+" "+
publishInfoDo.getRegionName()+" "+
publishInfoDo.getOwner()+" "+
publishInfoDo.getProducerId()+" "+
publishInfoDo.getTopic()+" "+
publishInfoDo.getStatus()+" "+
publishInfoDo.getStatusName()+" "+
publishInfoDo.getCreateTime()+" "+
publishInfoDo.getUpdateTime());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query PID

Description

OnsPublishGet interface queries detailed information of the specified PID.

Usage

The OnsPublishGet interface is called to display and query publishing information.

Request parameters

Name	Type	Required	Description

OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ProducerId	String	Yes	PID of the publishing relation to be queried.
Topic	String	Yes	The topic to be queried.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	A set of query results

PublishInfoDo data structure

Member	Type	Description
id	Long	ID of the publishing information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	The name of the region where the Topic is located. ALIYUN for Alibaba Cloud, CLOUD for Tmall Cloud...
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the

		OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for “publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status. (0 for “in service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status (“in service”, “frozen”, and “paused”)
appkey	Integer	null
createTime	Long	Creation time
updateTime	Long	Update time
remark	String	An optional note

Related APIs

- OnsPublishDelete: deletes a publishing relation
- OnsPublishList: retrieves a list of publishing relations
- OnsPublishCreate: creates a publishing relation

Example

This example shows the process of querying a PID named “PID_Mingduan” in the daily region, and printing the main attributes.

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
```

```
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsPublishGetRequest request = new OnsPublishGetRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
request.setProducerId("PID_Mingduan");
try {
OnsPublishGetResponse response=iAcsClient.getAcsResponse(request);
List<OnsPublishGetResponse.PublishInfoDo> publishInfoDoList =response.getData();
for (OnsPublishGetResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getId()+" "+
publishInfoDo.getChannelId()+" "+
publishInfoDo.getChannelName()+" "+
publishInfoDo.getOnsRegionId()+" "+
publishInfoDo.getRegionName()+" "+
publishInfoDo.getOwner()+" "+
publishInfoDo.getProducerId()+" "+
publishInfoDo.getTopic()+" "+
publishInfoDo.getStatus()+" "+
publishInfoDo.getStatusName()+" "+
publishInfoDo.getCreateTime()+" "+
publishInfoDo.getUpdateTime());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query PID

This topic introduces the OnsPublishSearch interface that queries publishing relations according to the search criteria.

Usage

When there are too many PIDs under a user account to present at once, you can call the OnsPublishSearch interface to search for publishing relations that match the keyword. The presented data format is consistent with that of the OnsPublishGet interface.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Search	String	Yes	A keyword for search, which can be Topic or PID.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(PublishInfoDo)	A set of query results

PublishInfoDo data structure

Member	Type	Description

id	Long	ID of the publishing information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	Number of the current status
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Topic	String	The Topic name
owner	String	The number of the Topic owner
relation	Integer	Numbers of all relations. 1 for the owner, 2 for “publish”, 4 for “subscribe”, and 6 for “publish and subscribe”.
relationName	String	Names of all relations such as the owner, “subscribe”, “publish”, and “publish and subscribe”.
status	Integer	Number of the current status. (0 for “in service”, 1 for “frozen”, and 2 for “paused”)
statusName	String	The alias of the current status (“in service”, “frozen”, and “paused”)
appkey	Integer	null
createTime	Long	Creation time
updateTime	Long	Update time
remark	String	An optional note

Related APIs

- OnsPublishDelete: deletes a publishing relation
- OnsPublishGet: queries a publishing relation
- OnsPublishCreate: creates a publishing relation

Example

This example is to search for information related to a PID in the daily region by taking PID_Mingduan as the search criterion and print the result.

This example shows the process of querying the PIDs with *PID_Mingduan* as the search keyword in the daily region, and printing the result.

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsPublishSearchRequest request = new OnsPublishSearchRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setSearch("PID_Mingduan");
try {
OnsPublishSearchResponse response=iAcsClient.getAcsResponse(request);
List<OnsPublishSearchResponse.PublishInfoDo> publishInfoDoList =response.getData();
for (OnsPublishSearchResponse.PublishInfoDo publishInfoDo:publishInfoDoList){
System.out.println(publishInfoDo.getId()+" "+
publishInfoDo.getChannelId()+" "+
publishInfoDo.getChannelName()+" "+
publishInfoDo.getOnsRegionId()+" "+
publishInfoDo.getRegionName()+" "+
publishInfoDo.getOwner()+" "+
publishInfoDo.getProducerId()+" "+
publishInfoDo.getTopic()+" "+
publishInfoDo.getStatus()+" "+
publishInfoDo.getStatusName()+" "+
publishInfoDo.getCreateTime()+" "+
publishInfoDo.getUpdateTime());
}
} catch (ServerException e) {
```

```
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Subscription management APIs

Create CID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsSubscribeCreate interface creates a subscription relation with a specified Topic on the server.

Usage

After a new application is enabled and the Topic resource is registered, to subscribe to messages, you must create and register the subscription relation with the target Topic in the MQ console or by calling this API, then you can use the registered CID to subscribe to messages.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.

PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	CID of the created message consumption cluster
Topic	String	Yes	The subscribed Topic

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsSubscribeDelete: deletes a subscription relation
- OnsSubscribeList: queries a list of subscription relations
- OnsSubscribeGet: precisely queries information of a subscription relation

Example

This example shows the process of creating a subscription relation “CID_MingduanTest” in the daily region to consume the messages of Mingduan_67dd.

```
 public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
```

```
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsSubscriptionCreateRequest request = new OnsSubscriptionCreateRequest();
/**
*OnsRegionId refers to the resource of the region of ONS required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("Mingduan_67dd");
request.setConsumerId("CID_MingduanTest");
try {
OnsSubscriptionCreateResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Delete CID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsSubscribeDelete interface deletes a subscription relation previously created with the OnsSubscribeCreate interface.

Usage

When an application is disabled and MQ resources need to be destroyed, you can call the OnsSubscribeDelete interface to delete the subscription relation.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It

			can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	CID of the message consumption cluster to be deleted
Topic	String	Yes	The subscribed Topic

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related interfaces

- OnsSubcribbeCreate: creates a subscription relation
- OnsSubscribeList: queries a list of subscription relations

Example

```
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
```

```
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsSubscriptionDeleteRequest request = new OnsSubscriptionDeleteRequest();
/**
*OnsRegionId refers to the resource of the region of ONS required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("Mingduan_67dd");
request.setConsumerId("CID_MingduanTest");
try {
OnsSubscriptionDeleteResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query CID

OnsSubscribeGet interface queries the detailed information of the target CID.

Usage

To manage the subscription relations and retrieve the detailed information of the CIDs under an account, you can call this interface.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.

OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	CID of the subscribe relation to be queried
Topic	String	Yes	Topic to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(SubscribeInfoDo)	The set of queried subscription relations

SubscribeInfoDo data structure

Name	Type	Description
Id	Long	The index number of the subscription information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	The name of the region where the Topic is located. ALIYUN for Alibaba Cloud, CLOUD for Tmall Cloud...
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Owner	String	Owner of the publishing information

ConsumerId	String	ID of a consumption cluster
Topic	String	The Topic name
Status	Long	The current status
StatusName	String	The status name, including In Service, offline, and so on.
CreateTime	Long	Creation time
UpdateTime	Long	The last update time

Related APIs

- OnsSubcribeCreate: creates a subscription relation
- OnsSubcribeDelete: deletes a subscription relation
- OnsSubscribeList: retrieves a list of subscription relations

Example

This example shows the process of querying and printing the information of a Consumer ID “CID_Mingduan” in the daily region.

```
  public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsSubscriptionGetRequest request = new OnsSubscriptionGetRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
```

```
request.setConsumerId("CID_Mingduan");
try {
OnsSubscriptionGetResponse response=iAcsClient.getAcsResponse(request);
List<OnsSubscriptionGetResponse.SubscribeInfoDo> subscribeInfoDoList=response.getData();
for(OnsSubscriptionGetResponse.SubscribeInfoDo subscribeInfoDo:subscribeInfoDoList){
System.out.println(subscribeInfoDo.getId()+" "+
subscribeInfoDo.getChannelId()+" "+
subscribeInfoDo.getChannelName()+" "+
subscribeInfoDo.getOnsRegionId()+" "+
subscribeInfoDo.getRegionName()+" "+
subscribeInfoDo.getOwner()+" "+
subscribeInfoDo.getConsumerId()+" "+
subscribeInfoDo.getTopic()+" "+
subscribeInfoDo.getStatus()+" "+
subscribeInfoDo.getStatusName()+" "+
subscribeInfoDo.getCreateTime()+" "+
subscribeInfoDo.getUpdateTime());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Fetch all CIDs of specified user

OnsSubscribeList interface retrieves a list of all CIDs of a user.

Usage

This interface presents a list of CIDs of a user instead of the detailed information of the CIDs.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.
OnsPlatform	String	No	The source of the request, which by default is POP

			platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(SubscribeInfoDo)	The set of queried subscribe relations

SubscribeInfoDo data structure

Name	Type	Description
Id	Long	The index number of the subscription information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	The name of the region where the Topic is located. ALIYUN for Alibaba Cloud, CLOUD for Tmall Cloud...
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Owner	String	Owner of the publishing information
ConsumerId	String	ID of a consumption cluster
Topic	String	The Topic name
Status	Long	The current status
StatusName	String	The status name, including In Service, offline, and so on.

CreateTime	Long	Creation time
UpdateTime	Long	The last update time

Related APIs

- OnsSubcribeCreate: creates a subscription relation
- OnsSubcribeDelete: deletes a subscription relation
- OnsSubscribeGet: precisely queries a subscription relation

Example

This example shows the process of querying and printing a list of all CIDs in the daily region.

```
 public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsSubscriptionListRequest request = new OnsSubscriptionListRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
try {
OnsSubscriptionListResponse response=iAcsClient.getAcsResponse(request);
List<OnsSubscriptionListResponse.SubscribeInfoDo> subscribeInfoDoList=response.getData();
for(OnsSubscriptionListResponse.SubscribeInfoDo subscribeInfoDo:subscribeInfoDoList){
System.out.println(subscribeInfoDo.getId()+" "+
subscribeInfoDo.getChannelId()+" "+
subscribeInfoDo.getChannelName()+" "+
subscribeInfoDo.getOnsRegionId()+" "+
subscribeInfoDo.getRegionName()+" "+
subscribeInfoDo.getOwner()+" "+
```

```
subscribeInfoDo.getConsumerId()+" "+
subscribeInfoDo.getTopic()+" "+
subscribeInfoDo.getStatus()+" "+
subscribeInfoDo.getStatusName()+" "+
subscribeInfoDo.getCreateTime()+" "+
subscribeInfoDo.getUpdateTime());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Search for CID

OnsSubscribeSearch interface searches for the matching CID information according to Topic or CID information.

Usage

This interface searches for CID information according to Topic when the name of CID is unknown.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Search	String	Yes	A keyword for search, which can be

			Topic or PID.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(SubscribeInfoDo)	The set of queried subscription relations

SubscribeInfoDo data structure

Name	Type	Description
Id	Long	The index number of the subscription information in the database
channelId	Integer	The ID of the region where the Topic is located, including 0-ALIYUN, 1-CLOUD, 2, 3, and 4
channelName	String	Number of the current status
onsRegionId	Long	The ID of the region where the Topic is located; obtained through the OnsRegionList method.
regionName	String	The name of the region where the Topic is located.
Owner	String	Owner of the publishing information
ConsumerId	String	ID of a consumption cluster
Topic	String	The Topic name
Status	Long	The current status
StatusName	String	The status name, including In Service, offline, and so on.
CreateTime	Long	Creation time
UpdateTime	Long	The last update time

Related APIs

- OnsSubcribeCreate: creates a subscription relation
- OnsSubcribeDelete: deletes a subscription relation
- OnsSubscribeList: retrieves a list of subscription relations

Example

This example shows the process of retrieving and printing the matching CID information with the search criterion of “CID_Mingduan”.

```
 public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsSubscriptionSearchRequest request = new OnsSubscriptionSearchRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setSearch("CID_Mingduan");
try {
OnsSubscriptionSearchResponse response=iAcsClient.getAcsResponse(request);
List<OnsSubscriptionSearchResponse.SubscribeInfoDo> subscribeInfoDoList=response.getData();
for(OnsSubscriptionSearchResponse.SubscribeInfoDo subscribeInfoDo:subscribeInfoDoList){
System.out.println(subscribeInfoDo.getId()+" "+
subscribeInfoDo.getChannelId()+" "+
subscribeInfoDo.getChannelName()+" "+
subscribeInfoDo.getOnsRegionId()+" "+
subscribeInfoDo.getRegionName()+" "+
subscribeInfoDo.getOwner()+" "+
subscribeInfoDo.getConsumerId()+" "+
subscribeInfoDo.getTopic()+" "+
subscribeInfoDo.getStatus()+" "+
subscribeInfoDo.getStatusName()+" "+
subscribeInfoDo.getCreateTime()+" "+
subscribeInfoDo.getUpdateTime());
```

```
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Configure CID permission

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsSubscribeUpdate interface sets the permission to read a message for a specified Consumer ID.

Usage

You can configure a switch of message reading for a Consumer ID determined by your specified MQ region and Consumer ID name. This is usually used for preventing a specific Consumer ID from reading messages.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.

ConsumerId	String	Yes	The name of Consumer ID to be configured
ReadEnable	Boolean	Yes	It sets a switch of message reading.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsSubcribeCreate: creates a subscription relation
- OnsSubcribeDelete: deletes a subscription relation
- OnsSubscribeGet: precisely queries a subscription relation

Example

As a reference, this example shows the process of accessing from the end point *China East*, and disabling the message reading permission of the Consumer ID “CID_001”.

```
    public static void main(String[] args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient IAcsClient= new DefaultAcsClient(profile);

OnsSubscriptionUpdateRequest request = new OnsSubscriptionUpdateRequest();
request.setAcceptFormat(FormatType.JSON);
request.setConsumerId("CID_001");
```

```
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setReadEnable(false);//2 indicates "write-only"; 4 indicates "read-only"; 6 indicates "read/write supported".
try {
OnsSubscriptionUpdateResponse response = IAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
}
catch (ServerException e) {
e.printStackTrace();
}
catch (ClientException e) {
e.printStackTrace();
}
catch (Exception e) {
e.printStackTrace();
}
}
```

Consumption management APIs

Query accumulation

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

This interface queries the message accumulation of a specified Consumer ID and retrieves the number of not consumed messages and an estimated delay time.

Usage

Consumer accumulate query is used when you need to understand the consumption progress of the Consumer ID in the production environment, and to have a rough idea about the message

consumption and delays.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	The consumer CID to be queried
Detail	Boolean	No	If to query detailed information. False by default.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	Data	The consumption accumulation of the specified Consumer

Data structure

Member	Type	Description
OnLine	Boolean	If the consumer is online
TotalDiff	Long	The total number of accumulated messages of all Topics that the current Consumer ID subscribes to

ConsumeTps	Float	TPS of current consumption
LastTimestamp	Long	The last update time
DelayTime	Long	The delay time
DetailInTopicList	List(OnsConsumerAccumulateResponse.Data.DetailInTopicDo)	Details of each Topic

DetailInTopicList data structure

Member	Type	Description
Topic	String	The Topic name
TotalDiff	Long	The consumption accumulation of the current Topic
LastTimestamp	Long	The last update time
TotalDiff	Long	The consumption delay time of the current Topic

Related APIs

- OnsConsumerStatus: queries the consumer status details
- OnsConsumerConnection: queries the consumer connection
- OnsResetOffset: clears the consumption accumulation

Example

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
```

```
OnsConsumerAccumulateRequest request = new OnsConsumerAccumulateRequest();
// request.setCluster("taobaodaily");
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setDetail(true);
request.setConsumerId("RTDSQ_1013_GROUP");
try {
OnsConsumerAccumulateResponse response=iAcsClient.getAcsResponse(request);
OnsConsumerAccumulateResponse.Data data =response.getData();
System.out.println(data.getOnline()+" "+data.getTotalDiff()+" "+data.getConsumeTps()+" "
+data.getDelayTime()+" "+data.getLastTimestamp());
for (OnsConsumerAccumulateResponse.Data.DetailInTopicDo detailInTopicDo:data.getDetailInTopicList()){
System.out.println(detailInTopicDo.getTopic()+" "
+detailInTopicDo.getTotalDiff()+" "
+detailInTopicDo.getLastTimestamp()+" "+detailInTopicDo.getDelayTime());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query consumer status

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

This interface queries the detailed status data of a specified Consumer ID, including subscription relation checking, consumption TPS count, server load balancer status, consumer connection, and so on.

Usage

This interface identifies the detailed status of the Consumer ID. It’ s called when you need to troubleshoot the exceptions after having a rough idea of the consumption accumulation and client

connection status. It tells if the subscription relations of a specified Consumer Group (CID) are consistent, if the server load balancer is normal, and the Jstack information of online clients.

Note: this interface calls a number of backend interfaces to aggregate data, and the query speed is relatively slow, so it’ s not recommended to call it frequently.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	The consumer CID to be queried
Detail	Boolean	No	If to query detailed information
NeedJstack	Boolean	No	If to print Jstack information

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	ConsumerStatusDo	Query result

ConsumerStatusDo data structure

Member	Type	Description
Online	Boolean	If it’ s online
TotalDiff	Long	The total consumption

		accumulation of a cluster
ConsumeTps	Long	TPS of the total consumption
LastTimestamp	Long	The last update time
DelayTime	Long	The delay time
ConsumeModel	Long	Consumption model
SubscriptionSame	Boolean	If the subscription relation is the same
RebalanceOK	Boolean	If the client Rebalance is normal
ConnectionSet	List(ConnectionDo)	The information of the current online client of the cluster
DetailInTopicList	List(DetailInTopicDo)	The consumption conditions of each Topic
ConsumerConnectionInfoList	List(ConsumerConnectionInfoDo)	The detailed information of the online client of the cluster, including Jstack, consumed RT time, and so on.

ConnectionDo data structure

Member	Type	Description
ClientId	String	ID of a consumption instance
ClientAddr	String	Address and port of the consumption instance
Language	String	Language of the consumer
Version	String	The consumer version

DetailInTopicDo data structure

Member	Type	Description
Topic	String	The Topic name
TotalDiff	Long	The total number of consumption accumulations of the Topic
LastTimestamp	Long	The last update time
DelayTime	Long	The delay time

ConsumerConnectionInfoDo data structure

Member	Type	Description
ClientId	String	ID of a consumption instance
Connection	String	Connection information
Language	String	Language of the client
Version	String	Version number of the client
ConsumeModel	String	The consume model, including Cluster and Broadcast.
ConsumeType	String	Where to consume
ThreadCount	Integer	The number of consumed threads
StartTimeStamp	Long	Start time
LastTimeStamp	Long	The last update time
SubscriptionSet	List(SubscriptionData)	The set of subscription relations
RunningDataList	List(ConsumerRunningDataDo)	Real-time status counting
Jstack	List(ThreadTrackDo)	Jstack stack information

SubscriptionData data structure

Member	Type	Description
Topic	String	The name of the subscribed Topic
SubString	String	The expression of the subcategory Tag subscribing the Topic.
SubVersion	Long	The version number of the subscription relation, which is incremental Long type.
TagsSet	List(String)	The set of subscribed Tags.

ConsumerRunningDataDo data structure

Member	Type	Description
ConsumerId	String	The Consumer ID name of the subscriber
Topic	String	The name of the subscribed

		Topic
Rt	Float	The RT consumption time in ms
OkTps	Float	TPS counting for messages consumed successfully
FailedTps	Float	TPS counting for messages failed to consume
FailedCountPerHour	Long	Count of messages failed to consume per hour

ThreadTrackDo data structure

Member	Type	Description
Thread	String	The thread name
TrackList	List(String)	Jstack stack information string

Related APIs

- OnsConsumerAccumulate: queries the consumption accumulation
- OnsConsumerConnection: queries the client connection of a consumer

Example

```
  public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsConsumerStatusRequest request = new OnsConsumerStatusRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
```

```
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setConsumerId("CID_Mingduan");
request.setDetail(true);
request.setNeedJstack(false);
try {
OnsConsumerStatusResponse response=iAcsClient.getAcsResponse(request);
OnsConsumerStatusResponse.Data data =response.getData();
List<OnsConsumerStatusResponse.Data.ConnectionDo> connectionDoList = data.getConnectionSet();
List<OnsConsumerStatusResponse.Data.DetailInTopicDo> detailInTopicDoList =data.getDetailInTopicList();
List<OnsConsumerStatusResponse.Data.ConsumerConnectionInfoDo> consumerConnectionInfoDoList
=data.getConsumerConnectionInfoList();
System.out.print(data.getOnline()+" "+data.getTotalDiff()+" "+data.getConsumeTps()+" "+
data.getLastTimestamp()+" "+data.getDelayTime()+" "+data.getConsumeModel()+
" "+data.getSubscriptionSame()+" "+data.getRebalanceOK());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query consumer connection

OnsConsumerConnection interface queries the connection of the current client in the specified Consumer ID (CID).

Usage

This interface tells if the consumer of the specified CID is online and retrieves a list of detailed client connections.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList

			method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	The consumer CID to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(ConnectionDo)	The connection information of the specified consumer

ConnectionDo data structure

Member	Type	Description
ClientId	String	ID of a consumption instance
ClientAddr	String	Address and port of the consumption instance
Language	String	Language of the consumer
Version	String	The consumer version

Related APIs

- OnsConsumerStatus: queries the consumption status
- OnsConsumerAccumulate: queries the consumption accumulation

Example

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
```

```
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsConsumerGetConnectionRequest request = new OnsConsumerGetConnectionRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setConsumerId("RTDSQ_1013_GROUP");
try {
OnsConsumerGetConnectionResponse response=iAcsClient.getAcsResponse(request);
List<OnsConsumerGetConnectionResponse.Data.ConnectionDo>
connectionDoList=response.getData().getConnectionList();
for(OnsConsumerGetConnectionResponse.Data.ConnectionDo connectionDo:connectionDoList){
System.out.println(connectionDo.getClientId()+" "+
connectionDo.getClientAddr()+" "+
connectionDo.getLanguage()+" "+
connectionDo.getVersion());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Reset consumer offset

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

This interface resets the consumer offset of the current Consumer ID to the specified time stamp according to the Consumer ID specified by the user. It’ s used for clearing accumulated messages or tracing back consumption. It provides two methods: one is to clear all messages, and the other is to clear the consumption progress to the specified time.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	The consumer CID to be queried
Topic	String	Yes	The name of the Topic of which the offset needs to be reset
Type	Integer	Yes	Set to0 to clear all messages, or set to 1 to clear the consumption progress to the specified time.
ResetTimestamp	Long	false	It resets the offset to the specified time stamp. Only effective when Type is set to 1.

Response parameters

Name	Type	Description

RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Example

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region where you are located, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName,regionId,productName,domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId,accessKey,secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsConsumerResetOffsetRequest request = new OnsConsumerResetOffsetRequest();
/**
*OnsRegionId refers to the resource of the region of ONS required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setConsumerId("CID_Mingduan");
request.setTopic("MingduanTest");
request.setType(1);
request.setResetTimestamp(System.currentTimeMillis());
try {
OnsConsumerResetOffsetResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query consumption progress

OnsConsumerTimeSpan interface queries the latest message time stamp of the Topic subscribed by the current Consumer ID, and the latest consumption time stamp.

Usage

This interface can be used for querying the latest and earliest time of a Topic’ s messages on the server, and querying the latest consumption time of the consumer. It’ s usually used in combination with the ConsumeAccumulate interface for getting a rough idea of the consumption progress.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
ConsumerId	String	Yes	The consumer CID to be queried
Topic	String	Yes	The Topic of which the consumption to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Data	Data	Query result

Data structure

Member	Type	Description
Topic	String	The name of the queried Topic.
MinTimeStamp	Long	The time of latest message stored in the Topic.
MaxTimeStamp	Long	The time of oldest message stored in the Topic.
ConsumeTimeStamp	Long	The time when the group last consumed the Topic.

Related APIs

- OnsConsumerAccumulate: queries the consumption accumulation
- OnsConsumerStatus: queries the consumer details
- OnsConsumerResetOffset: resets the offset and clears messages

Example

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsConsumerTimeSpanRequest request = new OnsConsumerTimeSpanRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
```

```
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setConsumerId("CID_Mingduan");
request.setTopic("MingduanTest12");
try {
OnsConsumerTimeSpanResponse response=iAcsClient.getAcsResponse(request);
OnsConsumerTimeSpanResponse.Data data =response.getData();
System.out.println(data.getTopic()+"\n"+
data.getConsumeTimeStamp()+"\n"+
data.getMaxTimeStamp()+"\n"+
data.getMinTimeStamp());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Message query APIs

Query message trace

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

Description

OnsMessageTrack interface determines if a message with the target MsgId has been consumed.

Usage

MessageTrack interface is usually used for determining if a specified message has been consumed, with the MsgId of a single message as the query criterion. Note that because this interface is implemented based on the internal offset system of MQ, the result is credible in most cases. However, if the user has reset the offset or cleared the messages, then the result can be incorrect.

Request parameters

Name	Type	Required	Description

OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Topic of the message
MsgId	String	Yes	ID of a message

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(MessageTrack)	Query result

MessageTrack data structure

Name	Type	Description
ConsumerGroup	String	The name of a consumption cluster
TrackType	String	Current status
ExceptionDesc	String	Exception description

Related APIs

- OnsMessageGetByMsgId: precisely queries a message by MsgId

Example

```
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsMessageTraceRequest request = new OnsMessageTraceRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
request.setMsgId("0A91883700001F90000001BF4723CB3E");
try {
OnsMessageTraceResponse response=iAcsClient.getAcsResponse(request);
List<OnsMessageTraceResponse.MessageTrack> trackList =response.getData();
for (OnsMessageTraceResponse.MessageTrack track:trackList){
System.out.println(track.getConsumerGroup()+" "+track.getTrackType()+" "+track.getExceptionDesc());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query message by message key

OnsMessageGetByMsgKey interface returns a list of matching messages by performing a fuzzy query with the transferred Topic and MsgKey.

Usage

Querying a message by MsgKey is a fuzzy query. Because the business party’ s keys may not be unique, there may be multiple query results. This interface is usually used when the business party cannot retrieve the MsgId, in which case, the business party needs to perform a fuzzy query to retrieve a list of MsgIds, then calls the OnsMessageGetByMsgId interface to query the message.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Topic of the message to be queried
Key	String	Yes	MsgKey of the message to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(OnsRestMessageDo)	Query result

OnsRestMessageDo data structure

Name	Type	Description
Topic	String	Topic of the message

Flag	Integer	null
PropertyList	List(MessageProperty)	List of message properties
Body	String	The message body
StoreSize	Integer	The message size
BornTimestamp	Long	The time stamp of generation
BornHost	String	Then client instance that generated this message
StoreTimestamp	Long	Time stamp stored by Broker
StoreHost	String	The server instance that stores the message
MsgId	String	Message ID
BodyCRC	Integer	CRC verification value of the message body
ReconsumeTimes	Integer	Times for re-consuming a message

Related APIs

- OnsMesageGetByMsgId: precisely queries a message by MsgId
- OnsMessageGetByTopic: queries a time range by Topic

Example

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsMessageGetByKeyRequest request = new OnsMessageGetByKeyRequest();
```

```
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
request.setKey("hello");
try {
OnsMessageGetByKeyResponse response = iAcsClient.getAcsResponse(request);
List<OnsMessageGetByKeyResponse.OnsRestMessageDo> onsRestMessageDoList=response.getData();
for(OnsMessageGetByKeyResponse.OnsRestMessageDo onsRestMessageDo:onsRestMessageDoList){
byte[] messageBody = Base64.decode(onsRestMessageDo.getBody());
String message =new String (messageBody);
System.out.println(onsRestMessageDo.getTopic()+" "+message+" "+
onsRestMessageDo.getFlag()+" "+
onsRestMessageDo.getBornHost()+" "+
onsRestMessageDo.getStoreSize()+" "+
onsRestMessageDo.getStoreHost()+" "+
onsRestMessageDo.getStoreTimestamp()+" "+
onsRestMessageDo.getReconsumeTimes());
}
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
```

Query message by message ID

OnsMessageGetByMsgId interface queries the information of the specified message with the transferred MsgId, including the sending time, storage server, and key and Tag of a message, and determines if the message has been consumed.

Usage

This function is used for determining if the message is successfully sent, and querying the sending time, target server, and so on.

Querying a message by MsgId is a precise query. MsgId required in the query criterion is retrieved from the SendResult after each successful sending . Therefore, the business party must store results of each sent message.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
MsgId	String	Yes	MsgId of the message to be queried
Topic	String	Yes	Topic of the message to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	Data	Query result

Data structure

Name	Type	Description
Topic	String	Topic of the message
Flag	Integer	null
PropertyList	List(MessageProperty)	List of message properties
Body	String	The message body
StoreSize	Integer	The message size
BornTimestamp	Long	The time stamp of generation

BornHost	String	Then client instance that generated this message
StoreTimestamp	Long	Time stamp stored by Server
StoreHost	String	The server instance that stores the message
MsgId	String	Message ID
BodyCRC	Integer	CRC verification value of the message body
ReconsumeTimes	Integer	Times for re-consuming a message

Related APIs

- OnsMessageGetByMsgKey: queries a message through a fuzzy matching by MsgKey
- OnsMessageGetByTopic: queries a message according to the Topic and time range

Example

```
  public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsMessageGetByMsgIdRequest request = new OnsMessageGetByMsgIdRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setMsgId("0ADA91A600002A9F000002ADDA8137E8");
request.setTopic("TopicTest");
```

```
try {
OnsMessageGetByMsgIdResponse response = iAcsClient.getAcsResponse(request);
OnsMessageGetByMsgIdResponse.Data data =response.getData();
byte[] msgbody= Base64.decode(data.getBody());
String message= new String(msgbody);
System.out.println(data.getTopic()+" "+
message+" "+
data.getFlag()+" "+
data.getBornHost()+" "+
data.getStoreSize()+" "+
data.getStoreHost()+" "+
data.getStoreTimestamp()+"  "+
data.getReconsumeTimes());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query message by Topic

OnsMessagePageQueryByTopic interface queries all messages within the specified time range by pages with transferred Topic and time range.

Usage

Querying a message by Topic is a range query. Query results are presented to the caller by pages. It is only applicable when no search criteria (MsgId or MsgKey) are available.

Note: It’ s recommended to limit the query time range as much as possible when querying by Topic. Otherwise, the issue cannot be identified due to the large amount of matching messages.

Calling method

- First, transfer the Topic, the start/end time, and page size, to query by pages. If there are messages, then it returns the messages on the first page, the total number of pages, and query Task ID by default.
- Then retrieve the messages according to the query Task ID in the returned result. When retrieving the messages, transfer the Task ID and the current number of pages.

Request parameters

Name	Type	Required	Description

OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Topic of the message to be queried
BeginTime	Long	Yes	The start time stamp of the query range
EndTime	Long	Yes	The end time stamp of the query range
TaskId	String	No	The query Task ID, not required for the first query, but required when retrieving messages later. It’s retrieved from the first return result.
CurrentPage	Integer	Yes	The number of the page where the messages are to be retrieved, incrementing from 1. It cannot exceed the maximum page number when retrieving the messages.
PageSize	Integer	No	The maximum amount of messages displayed on each page during query by pages, which by default is 20. The minimum value is 5, and the maximum value is 50.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
MsgFoundDo	MsgFoundDo	Query result

MsgFoundDo data structure

Name	Type	Description
TaskId	String	Task ID returned when query is created for the first time; used for retrieving the message later.
MaxPageCount	Long	The maximum number of pages of the query result
CurrentPage	Long	The current page number
Data	List(OnsRestMessageDo)	A set of messages on the current page. Its type is consistent with the return result of messageGetById.

Related APIs

- OnsMessageGetByMsgId: precisely queries a message by MsgId
- OnsMessageGetByMsgKey: fuzzily queries a message by MsgKey

Example

Example of creating query

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
```

```
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsMessagePageQueryByTopicRequest request = new OnsMessagePageQueryByTopicRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
request.setBeginTime(System.currentTimeMillis()-24*3600*1000);
request.setEndTime(System.currentTimeMillis());
request.setCurrentPage(1);
request.setPageSize(20);
try {
OnsMessagePageQueryByTopicResponse response = iAcsClient.getAcsResponse(request);
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Example of retrieving messages by pages

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXX";
String secretKey = "XXXXX";
IClientProfile profile = DefaultProfile.getProfile(regionId ,  accessKey ,  secretKey);
IAcsClient iAcsClient = new DefaultAcsClient(profile);
OnsMessagePageQueryByTopicRequest request = new OnsMessagePageQueryByTopicRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTest");
request.setBeginTime(System.currentTimeMillis()-24*3600*1000);
request.setEndTime(System.currentTimeMillis());
request.setCurrentPage(3);
request.setTaskId("0ADA91A600002A9F000002ADDA8137E8");
try {
```

```
OnsMessagePageQueryByTopicResponse response = iAcsClient.getAcsResponse(request);
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Message statistics APIs

Collect statistics on message production by Topic

OnsTrendTopicInputTps interface queries report data of message input for the specified Topic within a period of time.

Usage

This interface can be used for retrieving Topic message data in online environment. Depending on the type, you can query the Topic input amount in a specified time range, or you can query the TPS data in that time range.

If the application party has a small number of messages which are not evenly distributed, then TPS query may lead to scarce result. Instead, Total query is recommended.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP

			platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	The name of the Topic to be queried
BeginTime	Long	Yes	The start time stamp in milliseconds of the query range
EndTime	Long	Yes	The end time stamp in milliseconds of the query range
Period	Long	Yes	A sampling cycle in minutes, which can be 1, 5, or 10.
Type	Integer	Yes	The query type (0 for total, and 1 for TPS)

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
data	Data	A data set

Data set definition

Name	Type	Description
Title	String	The table name
Records	List(StatsDataDo)	Information of a collection point

StatsDataDo data set definition

Name	Type	Description
X	Long	The horizontal axis with time stamps in milliseconds
Y	Float	The vertical axis for the data (TPS or total)

Related APIs

- OnsTrendGroupOutputTps: queries the statistics of the message consumption of a CID

Example

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTrendTopicInputTpsRequest request =new OnsTrendTopicInputTpsRequest();
request.setOnsRegionId("cn-qingdao-publictest");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTestAPI");
request.setBeginTime(System.currentTimeMillis()-4*3600*1000);
request.setEndTime(System.currentTimeMillis());
request.setPeriod(1);
request.setType(0);
try {
OnsTrendTopicInputTpsResponse response =iAcsClient.getAcsResponse(request);
OnsTrendTopicInputTpsResponse.Data data =response.getData();
System.out.println(data.getTitle()+"\n"+
data.getRecords());
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Collect statistics on message consumption by CID

OnsTrendGroupOutputTps interface queries statistical information of a consumed message obtained by the specified Consumer ID within a period of time.

Usage

When generating consumption data report for online environment, you can use this interface to query the amount or TPS statistics of the consumed messages.

Query scenario:

- Query the curves of the total consumption
- Query the curves of the consumption TPS

If the application party has a small number of messages which are not evenly distributed, then TPS query may lead to scarce result. Instead, Total query is recommended.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	The name of the Topic to be queried
ConsumerId	String	Yes	The name of CID to be queried
BeginTime	Long	Yes	The start time stamp in milliseconds of the query range
EndTime	Long	Yes	The end time stamp in milliseconds of the query range

Period	Long	Yes	A sampling cycle in minutes, which can be 1, 5, or 10.
Type	Integer	Yes	The query type (0 for total, and 1 for TPS)

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
data	Data	A data set

Data set definition

Name	Type	Description
Title	String	The table name
Records	List(StatsDataDo)	Information of a collection point

StatsDataDo data set definition

Name	Type	Description
X	Long	The horizontal axis with time stamps in milliseconds
Y	Float	The vertical axis for the data (TPS or total)

Related APIs

- OnsTrendTopicInputTps: queries the statistics of message input

Example

```
    public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
```

```
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTrendGroupOutputTpsRequest request =new OnsTrendGroupOutputTpsRequest();
request.setOnsRegionId("cn-qingdao-publictest");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("MingduanTestAPI");
request.setConsumerId("CID_Mingduan");
request.setBeginTime(System.currentTimeMillis()-4*3600*1000);
request.setEndTime(System.currentTimeMillis());
request.setPeriod(1);
request.setType(0);
try {
OnsTrendGroupOutputTpsResponse response =iAcsClient.getAcsResponse(request);
OnsTrendGroupOutputTpsResponse.Data data =response.getData();
System.out.println(data.getTitle()+"\n"+
data.getRecords());
} catch (ClientException e) {
e.printStackTrace();
}
```

Resource authorization APIs

Create permission

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase.

Description

OnsEmpowerCreate interface authorizes specified users to use the target Topic, including the permission to publish and subscribe to messages.

Usage

MQ Topic resources are owned by the owner account that creates them. When it is required to access Topic resources across the account, the owner account can authorize the target account to publish or receive messages through an authorization interface.

Authorization considerations:

- Authorization supports three permissions: publish, subscribe, and publish & subscribe.
- The account which initiates authorization must be the resource owner account, and the target account can be a sub-account or primary account.
- After being authorized, the target account still needs to receive a message by creating its own subscription CID instead of using the CID of the owner account.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts, and cannot be used by RAM sub-accounts.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
EmpowerUser	Long	Yes	The authorized target user ID; supports sub-account ID
Topic	String	Yes	The authorized target Topic which must be owned by

			the current user.
Relation	Integer	Yes	The authorization type: 2 for “authorize to send”; 4 for “authorize to subscribe”, and 6 for “authorize to send and subscribe”.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsEmpowerList: views the authorization list
- OnsEmpowerDelet: deletes authorization

Example

```
    public static void main(String []args) {
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsEmpowerCreateRequest request = new OnsEmpowerCreateRequest();
request.setOnsRegionId("cn-qingdao-publictest");
```

```
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setEmpowerUser("167535647680831312");
request.setTopic("Topic_for_public_msg_test");
request.setRelation(2);
try {
OnsEmpowerCreateResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
}
```

View permission

OnsEmpowerList interface queries the authorization information of the target Topic on the specified account.

Usage

The Topic owner queries if a specified account is authorized to use a Topic he or she owns.The authorization type can be queried with this interface.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts, and cannot be used by RAM sub-accounts.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by

			default is POP platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
EmpowerUser	Long	Yes	The queried target user ID; supports sub-account ID
Topic	String	No	The queried target Topic which must be owned by the current user.

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
Data	List(AuthOwnerInfoDo)	The queried permission data list

AuthOwnerInfoDo structure

Name	Type	Description	
Topic	String	The Topic name	
Owner	Long	Owner UserId	
Relation	Integer	Yes	The authorization type: 2 for “authorize to send”; 4 for “authorize to subscribe”, and 6 for “authorize to send and subscribe”.

Related APIs

- OnsEmpowerCreate: creates authorization
- OnsEmpowerDelete: deletes authorization

Example

```
    public static void main(String []args) {
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsEmpowerListRequest request = new OnsEmpowerListRequest();
request.setOnsRegionId("cn-qingdao-publictest");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setEmpowerUser("1675352326808602");
request.setTopic("Topic_for_public_msg_test");
try {
OnsEmpowerListResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
}
```

Delete permission

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase.

Description

OnsEmpowerDelete interface deletes a previously created authorization, to remove the created

permission to access resources across accounts.

Usage

After creating authorizations, the owner account can call this interface to remove the Topic access permission across accounts.

NOTE:

- The caller must be the resource owner account.
- The authorization to be deleted must exist.

Restrictions on primary accounts and sub-accounts

This interface is exclusively available for primary accounts, and cannot be used by RAM sub-accounts.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
EmpowerUser	Long	Yes	The authorized target user ID; supports sub-account ID
Topic	String	Yes	The authorized target Topic which must be owned by the current user.

Response parameters

Name	Type	Description

RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link

Related APIs

- OnsEmpowerCreate: creates authorization
- OnsEmpowerList: views the authorization list

Example

```
public static void main(String []args) {
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsEmpowerDeleteRequest request = new OnsEmpowerDeleteRequest();
request.setOnsRegionId("cn-qingdao-publictest");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setEmpowerUser("16753563236808602");
request.setTopic("Topic_for_public_msg_test");
try {
OnsEmpowerDeleteResponse response=iAcsClient.getAcsResponse(request);
System.out.println(response.getRequestId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
}
```

Message tracing APIs

Obtain query result by task ID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

OnsTraceGetResultTest interface retrieves the previous trace query result by Task ID of the transferred trace query task.

Usage

This interface is used when the business party has created the trace query task by Message ID or Message Key and retrieved the query task ID. If the trace result details need to be retrieved with this Task ID, this interface can be used. Because the message tracing task often takes a while, retrieving results with this interface immediately after the task is created may lead to empty result, in which case, it’ s recommended to try again later.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	It specifies the region to be queried. The value is obtained through the OnsRegionList interface.
OnsPlatform	String	No	The source of the request, which by default is POP platform
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
QueryId	String	Yes	ID of the task of which the trace query result is to be queried

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
TraceData	TraceData	The result details of the query task

TraceData data structure

Name	Type	Description
queryId	String	ID of the query task
userId	String	ID of the query user
Topic	String	Topic of the query task
msgId	String	Message ID corresponding to the query task
msgKey	String	Message Key corresponding to the query task
status	String	The status of the query task, including “finish”, “working”, and “removed”.
createTime	Long	The creation time of the query task
updateTime	Long	Last update time of the query task
traceList	List(TraceMapDo)	List of matching traces of the query task

TraceMapDo data structure

Name	Type	Description
pubTime	Long	Message sending time
Topic	String	Topic of the message
pubGroupName	String	Producer ID configured by the sending client
msgId	String	Message ID of a message
Tag	String	Tag of a message

msgKey	String	Key of a message
bornHost	String	The client address of the message sender
costTime	Integer	Sending cost time in milliseconds
status	String	The sending status, including “SEND_SUCCESS”, “SEND_FAILED”, “SEND_ROLLBACK”, “SEND_UNKNOWN”, and “SEND_DELAY”.
subList	List(SubMapDo)	List of consumption traces of a message

SubMapDo data structure

Name	Type	Description
subGroupName	String	Consumer ID of a consumer
successCount	Integer	A count of successful consumption of the Consumer ID
failCount	Integer	A count of failed consumption of the Consumer ID
clientList	List(SubClientInfoDo)	List of client consumption details of the Consumer ID

SubClientInfoDo data structure

Name	Type	Description
subGroupName	String	Consumer ID to which the client belongs
subTime	Long	The consumption starting time stamp
clientHost	String	The client address of the consumer client
reconsumeTimes	Integer	The delivery round of the local consumption
costTime	Integer	Cost time in milliseconds of the current consumption
status	String	The consumption status, including

		“CONSUME_FAILED”, “CONSUME_SUCCESS”, and “CONSUME_NOT_RETURN”.

Related APIs

- OnsTraceQueryByMsgId: precisely queries the message trace by Message ID.
- OnsTraceQueryByMsgKey: fuzzily queries the message trace by Message Key.

Example

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTraceGetResultRequest request = new OnsTraceGetResultRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setQueryId("XXXXX");
try {
OnsTraceGetResultResponse response = iAcsClient.getAcsResponse(request);
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query trace by message ID

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

OnsTraceQueryByMsgId interface creates a trace query task with the transferred Topic and Message ID and returns the Task ID of the query task.

Usage

This interface can be used to create a query task when the business party has recorded the Message ID of a message and needs to query the delivery trace of this message with the Message ID. After the Task ID is returned by the interface, the OnsTraceGetResult interface can be used for retrieving the query result with the transferred Task ID.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP platform.
PreventCache	Long	Yes	It’ s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Topic of the message to be queried
MsgId	String	Yes	Message ID of the message to be queried
BeginTime	Long	Yes	The start time of the query time range

EndTime	Long	Yes	The end time of the query time range

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
QueryId	String	Task ID of the query task; the OnsTraceGetResult interface is called to retrieve the detailed result by the ID

Related APIs

- OnsTraceQueryByMsgKey: fuzzily queries the message trace by Message Key.
- OnsTraceGetResult: retrieves trace information by query Task ID.

Example

```
   public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";

/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTraceQueryByMsgIdRequest request = new OnsTraceQueryByMsgIdRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
```

```
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("XXXXX");
request.setMsgId("XXXXX");
request.setBeginTime(System.currentTimeMillis()-1000*3600*24);
request.setEndTime(System.currentTimeMillis());
try {
OnsTraceQueryByMsgIdResponse response = iAcsClient.getAcsResponse(request);
System.out.print(response.getQueryId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```

Query trace by message key

This interface is exclusive for MQ Platinum edition users. For more information, go to Platinum Edition Purchase Page.

OnsTraceQueryByMsgKey creates a trace query task with the transferred Topic and Message Key and returns the Task ID of the query task.

Usage

This interface can be used to create a query task when the business party has recorded the Message Key of a message and needs to query the delivery trace of this message with the key. After the Task ID is returned by the interface, the OnsTraceGetResult interface can be used for retrieving the query result with the transferred Task ID.

Request parameters

Name	Type	Required	Description
OnsRegionId	String	Yes	The region where the queried MQ is currently located. It can be obtained through the OnsRegionList method.
OnsPlatform	String	No	The source of the request, which by default is POP

			platform.
PreventCache	Long	Yes	It’s used for CSRF verification. Set it to the current system time.
Topic	String	Yes	Topic of the message to be queried
MsgKey	String	Yes	Message Key of the message to be queried
BeginTime	Long	Yes	The start time of the query time range
EndTime	Long	Yes	The end time of the query time range

Response parameters

Name	Type	Description
RequestId	String	A public parameter unique for each request
HelpUrl	String	A help link
QueryId	String	Task ID of the query task; the OnsTraceGetResult interface is called to retrieve the detailed result by the ID

Related APIs

- OnsTraceQueryByMsgId: precisely queries the message trace by Message ID.
- OnsTraceGetResult: retrieves trace information by query Task ID.

Example

```
public static void main(String []args) {
String regionId = "cn-hangzhou";
String accessKey = "XXXXXXXXXXXXXXXXX";
String secretKey = "XXXXXXXXXXXXXXXXX";
String endPointName ="cn-hangzhou";
String productName ="Ons";
String domain ="ons.cn-hangzhou.aliyuncs.com";
```

```
/**
*Select Region based on the region you will access, and set the corresponding access point.
*/
try {
DefaultProfile.addEndpoint(endPointName , regionId , productName , domain);
} catch (ClientException e) {
e.printStackTrace();
}
IClientProfile profile= DefaultProfile.getProfile(regionId , accessKey , secretKey);
IAcsClient iAcsClient= new DefaultAcsClient(profile);
OnsTraceQueryByMsgKeyRequest request = new OnsTraceQueryByMsgKeyRequest();
/**
*OnsRegionId refers to the resource of the region of MQ required to be accessed by API.
*This value must be selected and configured by the list obtained through the OnsRegionList method because
OnsRegionId is changing, and cannot be written as a fixed value.
*/
request.setOnsRegionId("daily");
request.setPreventCache(System.currentTimeMillis());
request.setAcceptFormat(FormatType.JSON);
request.setTopic("XXXXX");
request.setMsgKey("XXXXX");
request.setBeginTime(System.currentTimeMillis()-1000*3600*24);
request.setEndTime(System.currentTimeMillis());
try {
OnsTraceQueryByMsgKeyResponse response = iAcsClient.getAcsResponse(request);
System.out.print(response.getQueryId());
} catch (ServerException e) {
e.printStackTrace();
} catch (ClientException e) {
e.printStackTrace();
}
}
```