Alibaba Cloud CLI

Best Practices

Best Practices

Installation instructions of Alibaba Cloud CLI for RDS

For Windows system

Alibaba Cloud CLI is a management tool created based on Alibaba Cloud Open APIs. With this tool, you can easily call Alibaba Cloud Open APIs and manage the RDS product.

Background information

Alibaba Cloud CLI is compiled based on Python and needs to run in Python environment. It is recommended to install Alibaba Cloud CLI using pip, which can be applied to Windows, Linux, and Mac OS.

For more information about Python, go to https://www.python.org/. For more information about pip, go to https://pip.pypa.io/.

Prerequisites

Ensure that you have downloaded the Python 2.7.x installation package for Windows.

Go to Python official website (https://www.python.org/downloads/) to download Python 2.7.x installation package.

Ensure that your device has connected to the Internet.

Installation procedure

Install Python by following its installation wizard.

Notice: When you install Python, ensure to enable pip. Otherwise, you can not use pip to install Alibaba Cloud CLI.

If pip version 7.x or later is installed on your device, skip to step 3.

If the version of pip installed on your device is too old, the installation of Alibaba Cloud CLI will fail. You can use the following command to upgrade pip first and then do as the follow-on steps. Use pip version 7.x or later.

Run the following command to upgrade pip.

```
cd C:\Python27
python -m pip install -U pip
```

If the following information is displayed, pip is successfully upgraded.

```
Successfully uninstalled pip-7.1.2
Successfully installed pip-8.1.2
```

Run the following command to install Alibaba Cloud CLI.

```
cd C:\Python27
cd Scripts
pip install aliyuncli
```

If the following information is displayed, Alibaba Cloud CLI is successfully installed.

```
Successfully installed aliyuncli-2.1.2 colorama-0.3.3 jmespath-0.7.1
```

Run the following command to install RDS SDK.

```
cd C:\Python27\Scripts
pip install aliyun-python-sdk-rds
```

If the following information is displayed, RDS SDK is successfully installed.

```
Successfully installed aliyun-python-sdk-rds-2.0.3
```

If your RDS SDK is not the latest version, run the following command to upgrade the

existing RDS SDK.

```
pip install --upgrade aliyun-python-sdk-rds
```

Run the following command to verify the installation result.

```
cd C:\Python27\Scripts
aliyuncli rds help
```

If the help information is displayed, RDS SDK is successfully installed.

Note: you can also run the command aliyuncli rds help to query other commands, which can be used to verify the configuration result.

To create an access key, log on to the console of Alibaba Cloud official website (https://www.aliyun.com/), click AccessKeys and then create an access key according to the prompts.

Run the following command to configure the global parameters of Alibaba Cloud CLI.

```
cd C:\Python27\Scripts
aliyuncli configure
```

If the following information is displayed, input the parameters listed.

```
Aliyun Access Key ID [None]: <Enter Access Key ID>
Aliyun Access Key Secret [None]: <Enter Access Key Secret>
Default Region Id [None]: <Enter the RegionId of your instance, such as **cn-hangzhou**. Run aliyuncli rds
DescribeRegions --output table to query the region list.>
Default output format [None]: <Enter your desired output format, json, text or table.>
```

Run the following command to verify the configuration.

```
cd C:\Python27\Scripts
aliyuncli rds DescribeAccounts --DBInstanceId xxxxx
```

If the information about your instance account is displayed, it indicates that your instance has connected to Alibaba Cloud.

For Linux/UNIX/Mac OS systems

Alibaba Cloud CLI is a management tool created based on Alibaba Cloud Open APIs. With this tool, you can easily call Alibaba Cloud Open APIs and manage the RDS product.

Background information

Alibaba Cloud CLI is compiled based on Python and needs to run in Python environment. It is recommended to install Alibaba Cloud CLI using pip, which can be applied to Windows, Linux, and Mac OS.

For more information about Python, go to https://www.python.org/. For more information about pip, go to https://pip.pypa.io/.

Prerequisites

Ensure that you have downloaded the Python 2.7.x installation package for Windows.

Go to Python official website (https://www.python.org/downloads/) to download Python 2.7.x installation package.

Ensure that your device has connected to the Internet.

Installation procedure

Install Python according to its installation wizard.

When installing Python, ensure to enable pip. Otherwise, you can’ t use pip to install Alibaba Cloud CLI.

If pip version 7.x or later is installed on your device, skip to step 3.

If the version of pip installed on your device is too old, the installation of Alibaba Cloud CLI will fail. You can use the following command to upgrade pip first and then do as the follow-on steps. Use pip version 7.x and later.

Run the following command to upgrade pip.

```
pip install -U pip
```

If the following information is displayed, pip is successfully upgraded.

```
Successfully uninstalled pip-7.1.2
Successfully installed pip-8.1.2
```

Run the following command to install Alibaba Cloud CLI.

```
sudo pip install aliyuncli
```

If the following information is displayed, Alibaba Cloud CLI is successfully installed.

```
Successfully installed aliyuncli-2.1.2 colorama-0.3.3 jmespath-0.7.1
```

Run the following command to install RDS SDK.

```
sudo pip install aliyun-python-sdk-rds
```

If the following information is displayed, RDS SDK is successfully installed.

```
Successfully installed aliyun-python-sdk-core-2.0.35 aliyun-python-sdk-rds-2.0.3
```

If your RDS SDK is not the latest version, run the following command to upgrade the existing RDS SDK.

```
sudo pip install --upgrade aliyun-python-sdk-rds
```

Run the following command to verify the installation result.

```
aliyuncli rds help
```

If the help information is displayed, RDS SDK is successfully installed.

Note: you can also run the command aliyuncli rds help to query other commands, which can be used to verify the configuration.

To create an access key, log on to the console of Alibaba Cloud official website (https://www.aliyun.com/), click AccessKeys and then create an access key according to the prompts.

Run the following command to configure the global parameters of Alibaba Cloud CLI.

```
sudo aliyuncli configure
```

If the following information is displayed, input the parameters listed.

```
Aliyun Access Key ID [None]: <Enter Access Key ID>
Aliyun Access Key Secret [None]: <Enter Access Key Secret>
Default Region Id [None]: <Enter the RegionId of your instance, such as **cn-hangzhou**. Run aliyuncli rds
DescribeRegions --output table to query the region list.>
Default output format [None]: <Enter your desired output format, json, text or table.>
```

Run the following command to verify the configuration.

```
aliyuncli rds DescribeAccounts --DBInstanceId xxxxx
```

If the information about your instance account is displayed, it indicates that your instance has connected to Alibaba Cloud.

RDS advanced commands instructions

Instance export function

For RDS products, Alibaba Cloud CLI supports exporting DB instances to files, which is an extended function of the tool. You can choose an instance ID to export the corresponding instance to a file. The exported file content is in the json format, which is convenient for you to view or use in programming. The exported file is also an import file template. You can personalize modifications according to the file, and then import the file to create multiple DB instances.

Command example:

```
aliyuncli rds ExportDBInstance --DBInstanceId XXXXX --filename test
```

DBInstanceId: This is a required parameter, indicating the ID of the instance to be exported.

filename: This is a required parameter, indicating the file to which of the instances is to be exported. It can be an absolute path or a single filename. If it is a single filename, the default path ~/.alicloudcli/ will be chosen.

Instance import function

For RDS products, Alibaba Cloud CLI supports instance creation via file import to easily create multiple identical DB instances. You can create multiple identical instances simply by specifying a template file and calling a command.

Command example:

```
aliyuncli rds ImportDBInstance --filename test --count 5
```

Parameters description:

filename: Template file, which can be a filename or an absolute path. If it is a single filename, the system will search the file under the ~/.alicloudcli/ path.

count: Indicates the number of DB instances to be created. This value of this parameter is a number. If this number exceeds the maximal value of instances that can be created at one time, a DB instance will be created by default. We strongly recommend that you use the file exported with the alicloudcli rds ExportDBInstance command as a template to edit. Do not create separate templates to avoid mistakes.

Note: For details on the numbers of instances that can be created at one time, see the relevant documents of each product.

Special notice:

Currently DB instances created through RDS open APIs can only be charged on the Pay-As-You-Go basis. Therefore, your template file will be created as Pay-As-You-Go instances.

OSS Alibaba Cloud CLI command instructions

The following provides instructions and examples for OSS commands.

Config

Command Instructions:

Config --host oss.aliyuncs.com --accessid accessid --accesskey accesskey --sts_token token is used to configure the default host, ID, and Key of Alibaba Cloud CLI. The default host is oss.aliyuncs.com. To access oss-internal.aliyuncs.com, you can add host oss-internal.aliyuncs.com.

Example:

```
aliyuncli oss Config --host oss.aliyuncs.com --accessid accessid --accesskey accesskey --sts_token token
```

GetAllBucket

Command Instructions:

GetAllBucket is used to display a bucket created by the user.

Example:

```
aliyuncli oss GetAllBucket
```

CreateBucket

Command Instructions:

CreateBucket oss://bucket --acl [acl] oss://bucket is used to create a bucket, where oss://bucket indicates the bucket. The acl parameter can be included or excluded.

Examples:

```
aliyuncli oss CreateBucket oss://myfirstbucket --acl public-read

aliyuncli oss CreateBucket oss://mysecondbucket --acl private

aliyuncli oss CreateBucket oss://mythirdbucket
```

DeleteBucket

Command Instructions:

DeleteBucket oss://bucket is used to delete a bucket.

Examples:

```
aliyuncli oss DeleteBucket oss://mybucket

aliyuncli oss DeleteBucket oss://myfirstbucket
```

DeleteWholeBucket

Danger: This command is very risky as it will erase all the data and the erased data cannot be recovered. Use it with caution.

Command Instructions:

DeleteWholeBucket oss://bucket is used to delete a bucket and its internal objects and multipart contents.

Example:

```
aliyuncli oss DeleteWholeBucket oss://mybucket
```

GetAcl

Command Instructions:

GetAcl oss://bucketis used to get the ACL of a bucket.

Example:

```
aliyuncli oss GetAcl oss://mybucket
```

SetAcl

Command Instructions:

SetAcl oss://bucket –acl [acl] is used to modify the ACL of a bucket. The ACL can only be one of the three: private, public-read, or public-read-write.

Example:

```
aliyuncli oss SetAcl oss://mybucket --acl private
```

List

Command Instructions:

List oss://bucket/[prefix] [marker] [delimiter] [maxkeys] is used to list objects in a bucket.

Examples:

```
aliyuncli oss List oss://mybucket/folder1/folder2

aliyuncli oss List oss://mybucket/folder1/folder2 maker1
```

MkDir

Command Instructions:

MkDir oss://bucket/dirname is used to create an object ending with “/ ” and with the size of 0.

Example:

```
aliyuncli oss MkDir oss://mybucket/folder
```

ListAllObject

Command Instructions:

ListAllObject oss://bucket/[prefix] is used to display all objects in a bucket. The prefix can be specified for displaying objects.

Examples:

```
aliyuncli oss ListAllObject oss://mybucket

aliyuncli oss ListAllObject oss://mybucket/testfolder/
```

DeleteAllObject

Caution: This command is very risky as it will erase all the data and the erased data cannot be recovered. Use it with caution.

Command Instructions:

DeleteAllObject oss://bucket/[prefix] is used to delete all objects in a bucket. The prefix can be specified for deleting objects.

Examples:

```
aliyuncli oss DeleteAllObject oss://mybucket

aliyuncli oss DeleteAllObject oss://mybucket/testfolder/
```

DownloadAllObject

Command Instructions:

DownloadAllObject oss://bucket/[prefix] localdir --replace false is used to download all objects in a bucket to a local directory and keep the directory structure. The prefix can be specified for downloading. “replace false” indicates that if a local file with the same name already exists, it will not be overwritten during the downloading. “replace true” indicates that the local file with the same name will be overwritten.

Examples:

```
aliyuncli oss DownloadAllObject oss://mybucket /tmp/folder

aliyuncli oss DownloadAllObject oss://mybucket /tmp/folder --replace=false

aliyuncli oss DownloadAllObject oss://mybucket /tmp/folder --replace=true
```

DownloadToDir

Command Instructions:

DownloadToDir oss://bucket/[prefix] localdir --replace=false is used to download all objects in a bucket to a local directory and keep the directory structure. The prefix can be specified for downloading. “replace=false” indicates that if a local file with the same name already exists, it will not be overwritten during the downloading. “replace=true” indicates that the local file with the same name will be overwritten. It achieves the same effect as downloadallobject.

Examples:

```
aliyuncli oss DownloadToDir oss://mybucket /tmp/folder

aliyuncli oss DownloadToDir oss://mybucket /tmp/folder --replace false

aliyuncli oss DownloadToDir oss://mybucket /tmp/folder --replace true
```

UploadObjectFromLocalDir

Command Instructions:

UploadObjectFromLocalDir localdir oss://bucket/[prefix] --check_point check_point_file --replace false --check_md5 false --thread_num 5 is used to upload files from a local directory to a bucket. For example, if localdir is set to /tmp/ where three files are contained, namely a/b, a/c, and a, then the files will correspond to oss://bucket/a/b, oss://bucket/a/c, and oss://bucket/a after being uploaded to the OSS. If the prefix is set to mytest, then the files will correspond to oss://bucket/mytest/a/b, oss://bucket/mytest/a/c, and oss://bucket/mytest/a after being uploaded to the OSS. check_point check_point_file is to specify files. After files are specified, osscmd will put the uploaded local files into check_point_file as time stamps, and the uploadfromdir command will compare the time stamps of the files being uploaded with those recorded in check_point_file. If there are changes, the files will be re-uploaded. Otherwise, the files will be skipped. There is no check_point_file by default.

Note: Because the check_point_file records all uploaded files, when there are many uploaded files, the check_point_file could be sizable.

Examples:

```
aliyuncli oss UploadObjectFromLocalDir /mytemp/folder oss://mybucket

aliyuncli oss UploadObjectFromLocalDir /mytemp/folder oss://mybucket --check_point_file
/tmp/mytemp_record.txt
```

Put

Command Instructions:

Put localfile oss://bucket/object --content-type [content_type]--headers "key1:value1，key2:value2" is used to upload a local file to a bucket. You can specify the content-type of the target object, or specify custom headers.

Examples:

```
aliyuncli oss Put myfile.txt oss://mybucket

aliyuncli oss Put myfile.txt oss://mybucket/myobject.txt

aliyuncli oss Put myfile.txt oss://mybucket/test.txt --content-type plain/text --headers  “x-oss-meta-des:test，x-oss-meta-location:CN”

aliyuncli oss Put myfile.txt oss://mybucket/test.txt --content-type plain/text
```

Get

Command Instructions:

Get oss://bucket/object localfile is used to download an object to a local file.

Example:

```
aliyuncli oss Get oss://mybucket/myobject /tmp/localfile
```

MultiGet

Command Instructions:

MultiGet oss://bucket/object localfile --thread_num 5 is used to download an object to a local file in the multi-thread way.

Examples:

```
aliyuncli oss MultiGet oss://mybucket/myobject /tmp/localfile

aliyuncli oss MultiGet oss://mybucket/myobject /tmp/localfile
```

Cat

Command Instructions:

Cat oss://bucket/object is used to read and directly print the content of an object. Do not use it when the object content is too big in size.

Example:

```
aliyuncli oss Cat oss://mybucket/myobject
```

Meta

Command Instructions:

Meta oss://bucket/object is used to read and print the meta information of an object. The meta information includes content-type, file length, custom meta, and so on.

Example:

```
aliyuncli oss Meta oss://mybucket/myobject
```

Copy

Command Instructions:

Copy oss://source_bucket/source_object oss://target_bucket/target_object --headers="key1:value1，key2:value2" is used to copy a source object from the source bucket to a target object in the target bucket.

Example:

```
aliyuncli oss Copy oss://bucket1/object1 oss://bucket2/object2
```

Delete

Command Instructions:

Delete oss://bucket/object is used to delete an object.

Example:

```
aliyuncli oss Delete oss://mybucket/myobject
```

SignUrl

Command Instructions:

SignUrl oss://bucket/object --timeout [timeout_seconds] is used to generate a URL containing signature, and specify the timeout duration. This command is applicable when specified objects in a private bucket are provided for others to access.

Example:

```
aliyuncli oss SignUrl oss://mybucket/myobject
```

Init

Command Instructions:

Init oss://bucket/object is used to initialize an Upload ID. This Upload ID can be used together with the MultiUpload command.

Example:

```
aliyuncli oss Init oss://mybucket/myobject
```

ListPart

Command Instructions:

ListPart oss://bucket/object --upload_id xxx is used to Display the uploaded parts under the Upload ID of the specified object. See OSS API documentation for related concepts. The Upload ID must be specified.

Example:

```
aliyuncli oss ListPart oss://mybucket/myobject --upload_id 75835E389EA648C0B93571B6A46023F3
```

ListParts

Command Instructions:

ListParts oss://bucke is used to display the incompleted multipart Upload ID and objects in a bucket. If the system prompts that a bucket is not empty when you are trying to delete it, you can use this command to check whether there are multi-part contents.

Example:

```
aliyuncli oss ListParts oss://mybucket
```

GetAllPartSize

Command Instructions:

GetAllPartSize oss://bucket is used to display the total size of parts in a bucket that are uploaded with the Upload ID.

Example:

```
aliyuncli oss GetAllPartSize oss://mybucket
```

Cancel

Command Instructions:

Cancel oss://bucket/object --upload_id xxx is used to terminate the Multipart Upload event corresponding to the Upload ID.

Example:

```
aliyuncli oss Cancel oss://mybucket/myobject --upload_id D9D278DB6F8845E9AFE797DD235DC576
```

MultiUpload

Command Instructions (a):

MultiUpload localfile oss://bucket/object is used to upload a local file to the OSS in the multipart way.

Example:

```
aliyuncli oss MultiUpload /tmp/localfile.txt oss://mybucket/object
```

Command Instructions (b):

MultiUpload localfile oss://bucket/object --upload_id xxx --thread_num 10 --max_part_num 1000 is used to upload a local file to the OSS in the multipart way. The number of divided parts of the local file is specified by max_part_num. This command will first determine whether the ETag of the corresponding parts of the Upload ID is consistent with the MD5 value of the local file. If yes, the upload will be skipped. So if an Upload ID is generated before use, it will be included as a parameter. Even if the upload fails, the upload can be resumed by executing the same multiupload command.

Examples:

```
aliyuncli oss MultiUpload /tmp/localfile.txt oss://mybucket/object --upload_id
D9D278DB6F8845E9AFE797DD235DC576

aliyuncli oss MultiUpload /tmp/localfile.txt oss://mybucket/object --thread_num 5

aliyuncli oss MultiUpload /tmp/localfile.txt oss://mybucket/object --max_part_num 100
```

UploadPartFromFile

Command Instructions:

UploadPartFromFile localfile oss://bucket/object --upload_id xx --part_number xxx is mainly used for testing and is not recommended.

UploadPartFromString

Command Instructions:

UploadPartFromString oss://bucket/object --upload_id xxx --part_number xxx --data xxx is mainly

used for testing and is not recommended.

Introduction to CDN commands

The following provides instructions and examples to illustrate CDN commands:

Type one: service operation commands

OpenCdnService

Command instructions:

This command is used to activate CDN. You can perform domain name operations only after you have activated the CDN service. A single user can activate the service only once. Before activating the service, your account must have passed real-name authentication.

Example:

```
aliyuncli cdn OpenCdnService --InternetChargeType PayByTraffic
```

DescribeCdnService

Command instructions:

This command is used to query the CDN service status. This includes the current billing type, service activation time, the billing type that will take effect next time, and the current service status.

Example:

```
aliyuncli cdn DescribeCdnService
```

ModifyCdnService

Command instructions:

This command is used to change the CDN service billing type.

Example:

```
aliyuncli cdn ModifyCdnService --InternetChargeType PayByTraffic
```

Type two: domain name operation commands

DescribeUserDomains

Command instructions:

This command is used to query all domain names and statuses under a user name. Domain name statuses include running (indicating normal state of the domain name service), OK, stopped, configuring, and configuration failed.

DescribeCdnDomainDetail

Command instructions:

This command is used to obtain the basic information of the specified CDN domain configuration.

AddCdnDomain

Command instructions:

This command is used to add CDN domain names. Only one CDN domain name can be submitted at a time. A single user can add up to 20 domain names.

Restrictions:

Before creating a CDN domain name, you must first activate the CDN service. The CDN domain name must already have been filed. If the origin site content is not on the Alibaba Cloud platform, it must be reviewed. The review will be completed in one business day.

StartCdnDomain

Command instructions:

This command is used to enable a disabled CDN domain name, changing the DomainStatus to online.

NOTE: If the account corresponding to the domain name is in arrears or the domain name is invalid, you cannot call this interface to enable the CDN domain name properly.

StopCdnDomain

Command instructions:

This command is used to deactivate a CDN domain name, changing the DomainStatus to offline.

NOTE: After a CDN domain name is deactivated, its information will be retained and the system will automatically perform back-to-source processing for requests to the CDN domain name. If, for the moment, you do not need the CDN domain name, we recommend using the StopCdnDomain interface to suspend it.

DeleteCdnDomain

Command instructions:

This command is used to delete the current CDN domain name. Only one CDN domain name can be submitted at a time. After DeleteCdnDomain is called successfully, all the records associated with the CDN domain name will be deleted. If you only wish to temporarily suspend use of the CDN domain name, we recommend using the StopCdnDomain interface.

Caution: Be careful when using this command (we suggest you restore the domain name A record at the DNS provider before deleting the domain name) to ensure the domain name can still be accessed after deletion.

Type three: refresh and push commands

RefreshObjectCaches

Command instructions:

This command is used to refresh the file content on a node. It refreshes the specified URL content to the Cache node. Only one URL can be submitted each time.

Restrictions: For a single ID, up to 2,000 URL push and refresh requests and up to 100 directory push and refresh requests can be submitted daily.

Refresh and push interfaces include the RefreshObjectCaches and PushObjectCache interfaces.

PushObjectCache

Command instructions:

This command is used to actively push content from the origin site to the L2 Cache node. Upon first access, you can directly hit cache so as to relieve pressure on the origin site.

Restrictions: For a single ID, up to 2,000 URL push and refresh requests can be submitted daily. You must note that no directory push request is supported currently. Refresh and push interfaces include the RefreshObjectCaches and PushObjectCache interfaces.

DescribeRefreshTasks

Command instructions:

This command is used to check whether push and refresh statuse has taken effect for the whole site.

Note: The push and refresh statuses can be queried by task ID or URL. If neither the taskid nor objectpath is specified, it will query the first page of data (20 records) in the past seven days by default. A taskid and objectpath can be specified at the same time, with a logical relationship of AND. Only the data in the past seven days can be queried.

Type four: resource monitoring commands

DescribeCdnMonitorData

Command instructions:

This command is used to obtain the domain name metric data with a minimum granularity of five minutes. This includes the request hit rate, bytes hit rate, QPS, traffic, and average response time.

NOTE: When StartTime and EndTime are not specified, data of the past 24 hours is read by default.

Query by specified start time and end time is also supported. The start time and end time must both be specified. Data of the latest 30 days can be obtained with one domain at most.

Type five: log obtaining command

DescribeCdnDomainLogs

Command instructions:

This command is used to obtain the address for downloading the original access log of the specified domain name.

Notice: Log content is retained for two weeks at most.

Complex commands instructions

RDS complex commands

The example of the DescribeSlowLogs command is as follows:

```
aliyuncli rds DescribeSlowLogs --DBInstanceId xxxxx --StartTime 2015-09-24Z --EndTime 2015-09-24Z --DBName hms --PageSize 30 --PageNumber 1
```

Notice: The incorrect time format cannot be identified. The PageSize parameter is used to specify the number of data items displayed on each page, and only several fixed values are allowed.

ECS complex commands

The examples of the AddBackendServers command are as follows. You need to notice the parameter forms following InstanceIds.

```
aliyuncli ecs DescribeInstances --RegionId cn-hangzhou --InstanceIds "['i-23hello']"

aliyuncli ecs DescribeInstances --RegionId cn-hangzhou --InstanceIds "['i-23hello' , 'i-34hello']"
```

SLB complex commands

The example of the AddBackendServers command is shown as follows. You need to notice the parameter forms following BackendServers.

```
aliyuncli slb AddBackendServers --LoadBalancerId 14fd07a7569-cn-ningxia-am7-c01 --BackendServers
"[{'ServerId':'i-21os1d7jr'} , {'ServerId':'i-21h2knaxy'}]"
```

The example of the RemoveBackendServers command is shown as follows. You need to notice the parameter forms following BackendServers.

```
aliyuncli slb RemoveBackendServers --LoadBalancerId 14ffe8a7a47-cn-hangzhou-dg-a01 --BackendServers "['i-
23glad1uz' , 'i-236tbrnzn']"
```

CMS complex commands

The example of the DescribeMetricDatum command is shown as follows.

```
aliyuncli cms DescribeMetricDatum --MetricName MySQL_CpuUsage --StartTime 2015-10-08T00:00:00Z --EndTime
2015-10-08T00:01:00Z --Dimensions "{instanceId:'rds2izeze2izeze'}" --Period 5m --NextToken 1 --Length 100
```

Notice:

> CMS has only one action. All parameter obtaining is determined according to the incoming parameters such as Dimensions “{instanceId:’ rds2izeze2izeze’ }”, and the incoming time is GTM time CMS.
>
> The above command returns both CPU and memory information. When a certain information item is needed, you can use the Dimensions “{instanceId:’ rds445qv67ce32y4v1gi’ , type:’ cpuusage’ }” parameter to filter.

Multiple profile usage

To use the corresponding configuration of profile, you only need to include the profile XXX parameter, for example:

```
aliyuncli ecs DescribeInstanceStatus
```

In this case, the default configuration is used, which is also the global configuration.

```
aliyuncli ecs DescribeInstanceStatus –-profile test
```

In this case, the configuration under test is used, which takes effect for one time.