Alibaba Cloud E-MapReduce

SDK Reference

Issue: 20190918

MORE THAN JUST CLOUD | Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. **You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.**
2. **No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.**
3. **The content of this document may be changed due to product version upgrades , adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.**
4. **This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults " and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity , applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use**

or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified , reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates . The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).
6. Please contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus , page names, and other UI elements.	Click OK.
`Courier font`	It is used for commands.	Run the `cd /d C:/windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*

Style	Description	Example
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich {stand | slave}`

Contents

1 E-MapReduce SDK release notes

E-MapReduce SDK release notes for each version.

 Note:

- emr-core: Supports the interaction between data sources in OSS and Hadoop/Spark. It exists in the cluster by default. You do not need to package emr-code as part of a job.
- emr-tablestore: Supports the interaction between data sources in Table Store and Hadoop/Hive/Spark. Package emr-tablestore in the JAR file.
- emr-mns_2.10/emr-mns_2.11: Supports Spark to read data sources in MNS. Package emr-mns_2.10/emr-mns_2.11 in the JAR file.
- emr-ons_2.10/emr-ons_2.11: Supports Spark to read data sources in Message Queue (MQ). Package emr-ons_2.10/emr-ons_2.11 in the JAR file.
- emr-logservice_2.10/emr-logservice_2.11: Supports Spark to read data sources in Log Service. Package emr-logservice_2.10/emr-logservice_2.11 in the JAR file.
- emr-maxcompute_2.10/emr-maxcompute_2.11: Supports Spark to read data sources in MaxCompute. Package emr-maxcompute_2.10/emr-maxcompute_2.11 in the JAR file.

```
<! -- Supports   interactio  n   with   data   sources   in   OSS -->
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - core </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    <! -- Supports   interactio  n   with   data   sources   in
 Table   Store
-->
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - tablestore </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    <! -- Supports   interactio  n   with   data   sources   in   MNS
 ,  MQ ,  Log   Service ,  and   MaxCompute  ( in   the   Spark   1 .
 x   environmen  t ) -->
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - mns_2 . 10 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - logservice  _2 . 10 </ artifactId >
        < version > 1 . 4 . 1 </ version >
```

```
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - maxcompute  _2 . 10 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - ons_2 . 10 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    <! -- Supports   interactio  n   with   data   sources   in   MNS
,  MQ ,  Log   Service ,  and   MaxCompute  ( in   the   Spark   2 .
x   environmen  t )-->
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - mns_2 . 11 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - logservice  _2 . 11 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - maxcompute  _2 . 11 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
    < dependency >
        < groupId > com . aliyun . emr </ groupId >
        < artifactId > emr - ons_2 . 11 </ artifactId >
        < version > 1 . 4 . 1 </ version >
    </ dependency >
```

- **v1.4.1**
  - **MaxCompute: Fixes the problem of truncation for the DATETIME values.**
  - **MaxCompute: Fixes the thread-safety problem for the SimpleDateFormat class.**
- **v1.4.0**
  - **MaxCompute: Adds implementation based on the DataSource class. Only versions 2.x and above of Spark are supported.**
  - **Log Service: Adds implementation based on Direct API. Only versions 2.x and above of Spark are supported.**
  - **OTS: Optimizes read and write operations.**
  - **Fixes the bug that the access key of a user is replaced by the access key of a cluster app role when reading data sources in Log Service.**
- **v1.3.2**
  - **Fixes some bugs in Table Store.**

- v1.3.1
  - Fixes the problem that NullPointerExceptions are thrown in some scenarios when Spark interacts with Log Service.
  - E-MapReduce SDK supports the Spark 2.x environments since this version.
- v1.3.0
  - Supports Hadoop MapReduce, Spark, SparkSQL, and Hive to read data in Table Store.
  - MNS and Log Service support the MetaService service provided by E-MapReduce . Based on the MetaService service, you can access data in MNS and Log Service without AccessKeys.
  - Upgrades some dependencies.
- v1.1.3.1

  SDK:
  - Fixes the problem of dependency conflict between MNS and Spark/Hadoop packages.
  - Fixes the problem that NullPointerExceptions are thrown in some scenarios when Spark Streaming interacts with MNS.
  - Fixes some bugs for the Python SDK.
  - Supports custom time and locations in the scenarios when Spark Streaming integrates with Loghub.
- Core
  - Fixes the problem that Hadoop does not support the Snappy native files. Currently, E-MapReduce supports processing the Snappy files that Log Service have archived to OSS.
  - Fixes the problem that Spark does not support the Snappy files.
  - Fixes the problem that OSS does not support the two algorithms of the OutputCommitter class in Apache Hadoop 2.7.2.
  - Optimizes the performance of Hadoop/Spark reading and writing data in OSS.
  - Fixes the problem that a Log4j exception is thrown when Spark prints a job.

- v1.1.2
  - Fixes the problem that the `Connection ClosedExce ption` is thrown when a job is reading data in OSS.
  - Fixes the problem that some Hadoop commands are not available when accessing OSS data sources.
  - Fixes the `java . text . ParseExcep tion` : Unparseable date problem.
  - Optimizes the support of emr-core for local debugging.
  - Interprets the "_$folder$" files created in the earlier versions as directory paths instead of regular files.
  - Adds a retry mechanism for Hadoop/Spark failing to read data in OSS.
- v1.1.1
  - Fixes the imbalance of disk usage caused by writing temporary files in OSS locally.
  - Removes the $_folder$ tag files created during OSS directory creation in a job execution.
- v1.1.0
  - Upgrades the LogHub SDK to 0.6.2. Abandons the Client DB mode and uses the Server DB mode instead.
  - Upgrades the OSS SDK to 2.2.0. Fixes the run-time exceptions caused by the bugs of the OSS SDK.
  - Adds support for MNS.
  - Compatibility.
    - For the versions 1.0.x SDKs.
      - Interface:
        - Compatible
      - Namespace:
        - Incompatible: Adjusts the package structure. Modifies the package name from com.aliyun to com.aliyun.emr.
  - Modifies the groupId of the project from com.aliyun to com.aliyun.emr. The modified dependency in the POM file is as follows:

```
< dependency >
      < groupId > com . aliyun . emr </ groupId >
      < artifactId > emr - sdk_2 . 10 </ artifactId >
```

```
        < version > 1 . 1 . 3 . 1 </ version >
    </ dependency >
```

- v1.0.5
  - Optimizes the LoghubUtils interface and parameter input.
  - Optimizes the output format of data in LogStore. Adds the *topic* and the *source* fields.
  - Adds the parameter configurations for the time interval of pulling data in LogStore. Parameter name: `spark . logservice . fetch . interval . millis` . Default value: 200. Unit: milliseconds.
  - Upgrades the ODPS SDK to 0.20.7-public.
- v1.0.4
  - Downgrades the dependency of Guava to 11.0.2 to avoid a conflict with the dependency of Guava in Hadoop.
  - The MapReduce task supports files greater than 5 GB.
- v1.0.3
  - Adds configuration parameters related to the OSS Client.
- v1.0.2
  - Fixes the bug that the OSS URIs are resolved incorrectly.
- v1.0.1
  - Optimize the settings for OSS URIs.
  - Adds support for MQ.
  - Adds support for Log Service.
  - Supports the Append Object feature of OSS.
  - Supports uploading data in OSS using the multipart upload API.
  - Supports copying data from OSS using the upload part copy mode.

2 Create an AccessKey

To operate the E-MapReduce service by using APIs or the SDK, you must have an AccessKey.

 Notice:
You must keep your AccessKey.

1. Log on to the Alibaba Cloud official site.
2. Click Console.
3. Click AccessKeys.
4. Create and manage your AccessKey.

3 Java SDK

3.1 Download SDKs

Alibaba Cloud SDKs for E-MapReduce consists of two parts. The common part of Alibaba Cloud Java SDK is involved in aliyun-java-sdk-core. The E-MapReduce part is involved in aliyun-java-sdk-emr. We recommend that you can use Maven to manage your projects.

Maven dependencies

```
< dependency >
          < groupId > com . aliyun </ groupId >
          < artifactId > aliyun - java - sdk - core </ artifactId >
          < version > 2 . 3 . 2 </ version >
     </ dependency >
     < dependency >
          < groupId > com . aliyun </ groupId >
          < artifactId > aliyun - java - sdk - ecs </ artifactId >
          < version > 2 . 2 . 3 </ version >
      </ dependency >
```

Download links

aliyun-java-sdk-core-2.3.9.jar

aliyun-java-sdk-emr-2.2.2.jar

3.2 Get started

This section describes how to quickly create clusters, jobs, and execution plans by using Java SDKs.

Note:

We recommend that you use OpenAPI Explorer to call and generate SDK sample codes. OpenAPI Explorer allows you to call APIs of cloud services, dynamically generate SDK sample codes, and quickly retrieve interfaces.

Prerequisites

You can create a Maven project and then add Maven dependencies as follows:

```
< dependency >
           < groupId > com . aliyun </ groupId >
           < artifactId > aliyun - java - sdk - core </ artifactId >
           < version > 2 . 3 . 9 </ version >
```

```
        </ dependency >
        < dependency >
             < groupId > com . aliyun </ groupId >
             < artifactId > aliyun - java - sdk - emr </ artifactId >
             < version > 2 . 2 . 2 </ version >
         </ dependency >
```

You can also download associated JAR files to your local disk. Take Eclipse as an example. You can download JAR files as follows:

1. Download the following JAR files:

   aliyun-java-sdk-core-2.3.9.jar

   aliyun-java-sdk-emr-2.2.2.jar
2. Copy the JAR files to your project folder.
3. In Eclipse, right-click your project name, and then select Properties > Java Build Path > Add JARs.
4. Select all JAR files that you copied in Step 2.

Now, you can use Alibaba Cloud E-MapReduce Open API Java SDK for your Eclipse project.

Initialize a client

```
IClientPro  file   profile  =  DefaultPro  file . getProfile (" cn -
hangzhou ", "< Your - AccessKeyI  d >", "< Your - AccessKeyS  ecret
>");
    DefaultAcs  Client   client  =  new   DefaultAcs  Client (
profile );
```

All operations on the E-MapReduce in SDK can be performed using this client.

Sample code

- Clusters
  - Create a cluster

```
public   static   void   main ( String []  args ) {
      IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
      DefaultAcs  Client   client  =  new   DefaultAcs  Client (
profile );
      final   CreateClus  terRequest   request  =  new
CreateClus  terRequest ();
      request . setName (" Your - Cluster - Name ");
     // if   you   did   not   specify   security   group   id
, it   will   create   a   new   security   group   with   given
  name
      request . setSecurit  yGroupId (" Your - Security - Group -
Id "); // ( 1 )
      request . setAutoRen  ew ( false );
```

```
      request . setChargeT  ype (" PostPaid "); //  PostPaid   or
  PrePaid
      request . setCluster  Type (" HADOOP "); //  HADOOP   or
HBase  ( 2 )
      request . setEmrVer (" EMR - 1 . 3 . 0 "); //  emr   image
  version
      request . setIsOpenP  ublicIp ( true );
      request . setLogEnab  le ( true );
      request . setLogPath (" oss :// Your - Bucket / Your -
Folder ");
      request . setMasterP  wdEnable ( true ); //  enable
master   password
      request . setMasterP  wd (" Aa12345678  9 "); //  set
master   node ' s   password
      request . setZoneId (" cn - hangzhou - b "); //  set
zone
      //  The   I / O   optimizati  on   parameters .  The
available   hardware   configurat  ions ,  such   as   ECS
instance   types   and   cloud   disk   types   are   determined
  by   the   specified   ECS   instance   series .
     //  For   more   informatio  n   about   available
configurat  ions ,  see   the   Buy   Now   page   of   ECS .
     //  https :// ecs . console . aliyun . com /#/ create /
postpay /
      request . setIoOptim  ized ( true ); //  You   can
specify   I / O   optimizati  on   parameters .
      request . setInstanc  eGeneratio  n (" ecs - 2 "); //  You
  can   specify   ecs - 2   as   an   ECS   instance   series .
Valid   values :  ecs - 1   and   ecs - 2 .
      request . setNetType (" classic "); //  You   can   specify
  classic   as   a   network   type .  Valid   values :  classic
  and   vpc .
      List < CreateClus  terRequest . EcsOrder >  ecsOrders  =
new   ArrayList < CreateClus  terRequest . EcsOrder >();
      CreateClus  terRequest . EcsOrder   masterOrde  r  =  new
CreateClus  terRequest . EcsOrder ();
      masterOrde  r . setIndex ( 1 );
      masterOrde  r . setDiskCap  acity ( 50 );
      masterOrde  r . setDiskCou  nt ( 2 );
      masterOrde  r . setDiskTyp  e (" CLOUD_EFFI  CIENCY "); //
specify   disk   type  ( 2 )
      masterOrde  r . setInstanc  eType (" ecs . n1 . large
"); //  specify   ecs   instance   type
      masterOrde  r . setNodeCou  nt ( 1 );
      masterOrde  r . setNodeTyp  e (" MASTER "); //  MASTER   or
  CORE  ( 2 )
      ecsOrders . add ( masterOrde  r );
      CreateClus  terRequest . EcsOrder   coreOrder  =  new
CreateClus  terRequest . EcsOrder ();
      coreOrder . setIndex ( 2 );
      coreOrder . setDiskCap  acity ( 50 );
      coreOrder . setDiskCou  nt ( 4 );
      coreOrder . setDiskTyp  e (" CLOUD_EFFI  CIENCY ");
      coreOrder . setInstanc  eType (" ecs . n1 . large ");
      coreOrder . setNodeCou  nt ( 3 );
      coreOrder . setNodeTyp  e (" CORE ");
      ecsOrders . add ( coreOrder );
      request . setEcsOrde  rs ( ecsOrders );
      try  {
          CreateClus  terRespons  e   response  =  client .
getAcsResp  onse ( request );
          String   clusterId  =  response . getCluster  Id
(); //  cluster   id
         //  TODO   do   something   with   this   cluster
```

```
        } catch ( Exception e ) {
            // TODO do something
        }
    }
```

- When you create a cluster, you must specify a security group that hosts this cluster. If you did not specify the ID of a security group, then you must specify the name of this security group. You need to create a security group when you create a cluster.
- For more information, see Enumeration.
- The preceding code snippet creates a cluster in a classic network. If you want to create a cluster in a VPC network, you need to specify vpc in the request method and specify vpcid and vswitchid as follows:

```
request . setNetType (" vpc "); // You can specify vpc
  as a network type .
request . setVpcId (" your - vpcId ");
request . setVSwitch Id (" your - switchId ");
```

- You can specify high availability parameters. For more information about high availability parameters, see the hardware configuration section of #unique_10.

```
request . setHighAva ilabilityE nable ( true );
```

- You can specify available software components. For more information about available software components, see the software configuration section of Create a cluster.

```
List < String > soft = new ArrayList < String >();
soft . add (" presto ");
soft . add (" oozie ");
request . setOptionS oftWareLis ts ( soft );
```

- You can specify a configuration item. For more information, click here.

```
Request . setconfigu rations (" Oss : // your - bucket /
your - conf . json ");
```

- You can specify a bootstrap operation. For more information, click here.

```
List < CreateClus terRequest . BootstrapA ction >
bootstrapA ctionLists = new ArrayList < CreateClus
terRequest . BootstrapA ction >();
CreateClus terRequest . BootstrapA ction bootstrapA
ctionList = new CreateClus terRequest . BootstrapA ction
();
bootstrapA ctionList . setName (" bootstrapN ame ");
bootstrapA ctionList . setPath (" oss :// emr - agent - pack
/ bootstrap / run - if . py ");
```

```
bootstrapA  ctionList . setArg (" instance . isMaster = true
mkdir  - p  / tmp / abc ");
bootstrapA  ctionLists . add ( bootstrapA  ctionList );
request . setBootstr  apActions ( bootstrapA  ctionLists );
```

- **Query the details of a cluster**

```
public   static   void   main ( String []  args ) {
      IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
      DefaultAcs  Client   client  =  new   DefaultAcs  Client (
profile );
      final   DescribeCl  usterReque  st   request  =  new
DescribeCl  usterReque  st ();
      request . setId (" C - XXXXXXXXXX  XXXXXX "); //  cluster
id
      try  {
          DescribeCl  usterRespo  nse   response  =  client .
getAcsResp  onse ( request );
          DescribeCl  usterRespo  nse . ClusterInf  o
clusterInf  o  =  response . getCluster  Info ();
         //  TODO   do   something   with   this   cluster
     }  catch  ( Exception   e ) {
         //  TODO   do   something
     }
 }
```

- **Query the list of clusters**

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          final   ListCluste  rsRequest   request  =  new
ListCluste  rsRequest ();
          request . setPageNum  ber ( 1 );
          request . setIsDesc ( true );
          request . setPageSiz  e ( 20 );
          try  {
              ListCluste  rsResponse   response  =  client .
getAcsResp  onse ( request );
              List < ListCluste  rsResponse . ClusterInf  o >
clusterInf  os  =  response . getCluster  s ();
              for  ( ListCluste  rsResponse . ClusterInf  o
clusterInf  o  :  clusterInf  os ) {
                 //  TODO   do   something   with   this
cluster
              }
          }  catch  ( Exception   e ) {
              //  TODO   do   something
          }
     }
```

- **Release a cluster**

```
public   static   void   main ( String []  args ) {
           IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your   AccessKeyI  d >", "<
Your   AccessKeyS  ecret >");
```

```
            DefaultAcs Client client = new DefaultAcs
Client ( profile );
            ReleaseClu sterReques t request = new
ReleaseClu sterReques t ();
            request . setId (" C - XXXXXXXXXX XXXXXX "); //
specify the cluster id you want to release .
            try {
                ReleaseClu sterRespon se response = client
. getAcsResp onse ( request );
            } catch ( Exception e ) {
                // TODO do something
            }
        }
```

- Jobs
  - Create a job

```
public static void main ( String [] args ) {
      IClientPro file profile = DefaultPro file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI d >", "<
Your - AccessKeyS ecret >");
      DefaultAcs Client client = new DefaultAcs Client (
profile );
      final CreateJobR equest request = new CreateJobR
equest ();
      request . setName (" Your - Job - Name ");
      request . setRunPara meter ("-- master yarn - client
-- driver - memory 4g -- executor - memory 4g -- executor
- cores 2 -- num - executors 4 -- class com . test .
RemoteDebu g ossref :// Your - Bucket / Resource . jar 1000
\"");
      request . setFailAct (" CONTINUE "); // STOP or
CONTINUE
      request . setType (" SPARK "); // SPARK or HADOOP or
  HIVE or PIG
```

```
try {
            CreateJobR esponse response = client .
getAcsResp onse ( request );
            String jobId = response . getId ();
          // TODO do something with this job
        } catch ( Exception e ) {
            // TODO do something
        }
    }
```

  - Delete a job

Notice:

When a job is being used by an execution plan, you cannot delete this job. You must delete this execution plan or modify this execution plan before you delete this job.

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

```
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          final   DeleteJobR  equest   request  =  new
DeleteJobR  equest ();
          request . setId (" J - XXXXXXXXXX  XXXXXX "); //  set
  job   id
          try  {
              DeleteJobR  esponse   response  =  client .
getAcsResp  onse ( request );
          }  catch  ( Exception   e ) {
              //  TODO   do   something
          }
      }
```

- **Execution plans**
  - **Creating an execution plan**

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          final   CreateExec  utionPlanR  equest   request  =
new   CreateExec  utionPlanR  equest ();
          request . setName (" Your - ExecutionP  lan - Name ");
          request . setCreateC  lusterOnDe  mand ( false );
          request . setStrateg  y (" RUN_MANUAL  LY "); //
RUN_MANUAL  LY   or   SCHEDULE
          request . setCluster  Id (" C - XXXXXXXXXX  XXXXXX
"); //  specify   an   existing   running   cluster
          List < String >  jobIds  =  new   ArrayList < String
>();
          jobIds . add (" J - XXXXXXXXXX  XXXXXX "); //  specify
  a   job
          request . setJobIdLi  sts ( jobIds );
          try  {
              CreateExec  utionPlanR  esponse   response  =
client . getAcsResp  onse ( request );
              String   executionP  lanId  =  response . getId ();
             //  TODO   do   something   with   this   execution
  plan
          }  catch  ( Exception   e ) {
              //  TODO   do   something
          }
      }
```

The preceding code snippet creates an execution plan with the RUN_MANUALLY type. In addition, an existing cluster is specified for this execution plan.

If you want to create an execution plan with the SCHEDULE type, you must modify the previous code snippet and add the following code snippet:

```
request . setStrateg  y (" SCHEDULE "); //  RUN_MANUAL  LY   or
SCHEDULE
```

```
          request . setStartTi  me ( new   Date (). getTime
()); //  set   start   time
          request . setTimeUni  t (" DAY "); //  DAY   or   HOUR
          request . setTimeInt  erval ( 1 ); //  set   time
interval
```

If you want to create an execution plan for which a new cluster will be created to run jobs, you must modify the previous code snippet and add the following code snippet:

```
request . setCreateC  lusterOnDe  mand ( true );
          request . setCluster  Type (" HADOOP ");
          request . setCluster  Name (" Your - Cluster - Name ");
          request . setEmrVer (" EMR - 1 . 3 . 0 ");
          request . setSecurit  yGroupId (" Your - Security -
Group - Id ");
          request . setIsOpenP  ublicIp ( true );
          //  The   I / O   optimizati  on   parameters .  The
  available   hardware   configurat  ions   such   as   ECS
instance   types   and   cloud   disk   types   are   determined
  by   the   specified   ECS   instance   series .
         //  For   more   informatio  n   about   how   to
select   these   configurat  ions ,  see   the   Buy   Now   page
  of   Elastic   Compute   Service .
         //  https :// ecs . console . aliyun . com /#/ create /
postpay /
          request . setIoOptim  ized ( true ); //  You   can
specify   true   to   enable   I / O   optimizati  on .
          request . setInstanc  eGeneratio  n (" ecs - 2 "); //
You   can   specify   esc - 2   as   an   ECS   instance   series
.  Valid   values :  ecs - 1   and   ecs - 2 .
          request . setNetType (" classic "); //  You   can
specify   classic   as   a   network   type .  Valid   values :
classic   and   vpc .
          request . setLogEnab  le ( true );
          request . setLogPath (" oss :// xxx ");
          request . setEcsOrde  rs (); //  TODO   you   can
  refer   to   the   configurat  ions   when   you   create
a   cluster .  Note   that   the   type   of   ecsOder   is
CreateExec  utionPlanR  equest . EcsOrder .  The   type   of
ecsOder   here   is   different   from   CreateClus  terRequest .
EcsOrder .
```

You can configure a cluster by specifying the previous parameters. For more information about these parameters, see the section Create a cluster. For an execution plan that requires a new cluster to run jobs, a temporary cluster will be created whenever you execute this execution plan. The temporary cluster is created based on the specified cluster configurations and then released after the execution plan is completed. Unlike the general process of creating a cluster,

you must specify a security group ID rather than a security group name when you create a cluster in this case.

The Create on demand option and the Periodic schedule option are not mutually exclusive. A new cluster is created when an execution plan starts at a scheduled time.

- Deleting an execution plan

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          final   DeleteExec  utionPlanR  equest   request  =
new   DeleteExec  utionPlanR  equest ();
          request . setId (" WF - XXXXXXXXXX  XXXXXX "); //  set
  execution   plan   id
          try  {
              DeleteExec  utionPlanR  esponse   response  =
client . getAcsResp  onse ( request );
          }  catch  ( Exception   e ) {
              //  TODO   do   something
          }
     }
```

- Execute an execution plan

Notice:

You cannot execute an execution plan that is running or being scheduled.

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your - AccessKeyI  d >", "<
Your - AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          RunExecuti  onPlanRequ  est   request  =  new
RunExecuti  onPlanRequ  est ();
          request . setId (" WF - XXXXXXXXXX  XXXXXX "); //
specify   the   execution   plan   id   which   to   run
          try  {
              RunExecuti  onPlanResp  onse   response  =  client
. getAcsResp  onse ( request );
              String   instanceId  =  response . getExecuti
onPlanInst  anceId ();
              //  TODO   do   something   with   this   instance
          }  catch  ( Exception   e ) {
              //  TODO   do   something
          }
```

```
    }
```

- **Suspend a scheduled execution plan**

  For an execution plan that will be executed periodically , you can suspend this execution plan by calling a method provided by the SDK as follows:

```
public   static   void   main ( String []  args ) {
          IClientPro  file   clientProf  ile  =  DefaultPro  file
. GetProfile (" cn - shanghai ", "< your - access - key - id >",
"< your - access - key - secret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          SuspendExe  cutionPlan  SchedulerR  equest   request
=  new   SuspendExe  cutionPlan  SchedulerR  equest ();
          request . setId (" WF - XXXXXXXXXX  XXXXXX "); //
specify   the   execution   plan   id   you   want   to
suspend
          try  {
              SuspendExe  cutionPlan  SchedulerR  esponse
response  =  client . getAcsResp  onse ( request );
         }  catch  ( Exception   e ) {
             //  TODO   do   something
         }
     }
```

- **Resume an execution plan**

  For an execution plan that will be executed periodically , you can resume this execution plan by calling a method provided by the SDK as follows:

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your   AccessKeyI  d >", "<
Your   AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          ResumeExec  utionPlanS  chedulerRe  quest   request  =
new   ResumeExec  utionPlanS  chedulerRe  quest ();
          request . setId (" WF - XXXXXXXXXX  XXXXXX "); //
specify   the   execution   plan   id   you   want   to
suspend
          try  {
              ResumeExec  utionPlanS  chedulerRe  sponse
response  =  client . getAcsResp  onse ( request );
         }  catch  ( Exception   e ) {
             //  TODO   do   something
         }
     }
```

- **Query the running logs of an execution plan**

```
public   static   void   main ( String []  args ) {
          IClientPro  file   profile  =  DefaultPro  file .
getProfile (" cn - hangzhou ", "< Your   AccessKeyI  d >", "<
Your   AccessKeyS  ecret >");
          DefaultAcs  Client   client  =  new   DefaultAcs
Client ( profile );
          ListExecut  ionPlanIns  tancesRequ  est   request  =
new   ListExecut  ionPlanIns  tancesRequ  est ();
```

```
        //  specify   execution   plan   ids
         List < String >  executionP  lanIds  =  new   ArrayList
< String >();
         executionP  lanIds . add (" WF - XXXXXXXXX  XXXXX1 ");
         executionP  lanIds . add (" WF - XXXXXXXXX  XXXXX2 ");
         executionP  lanIds . add (" WF - XXXXXXXXX  XXXXX3 ");
         request . setExecuti  onPlanIdLi  sts ( executionP
lanIds ); // ( 1 )
        //  specify   order   key  ( ordered   by   id )
         request . setIsDesc ( true );
        //  specify   page   number   and   page   size ,
default   page   number   is   1   and   default   page   size
is   10 .
         request . setPageSiz  e ( 20 );
         request . setPageNum  ber ( 1 );
        //  specify   if   you   want   to   list   latest
instance   for   each   execution   plan   id .
         request . setOnlyLas  tInstance ( true ); // ( 2 )
default   is   false
         try  {
             ListExecut  ionPlanIns  tancesResp  onse   response
 =  client . getAcsResp  onse ( request );
              for  ( ListExecut  ionPlanIns  tancesResp  onse .
ExecutionP  lanInstanc  e   instance  :  response . getExecuti
onPlanInst  ances ()) {
                  //  TODO   do   something   with   each
instance
              }
         }  catch  ( Exception   e ) {
             //  TODO   do   something
         }
     }
```

- You can specify multiple execution plan IDs to query the running logs.
- If you query the last running log, only the last running log of a specified execution plan will be returned. You can use this feature to check whether the last execution plan has been executed successfully, or query the last running log of an execution plan.

3.3 Reference project

If you are still not clear about how to construct a project through the previous steps, or you want to have a project that can be directly imported to an IDE for running, you can download the following reference project code and import it to your IDE environment.

Notice:

You need to change the information of the AccessKey and security group of this project to yours. For more information, see comments in the sample code.

Eclipse

Eclipse version used by the reference project: `Eclipse Java EE IDE for Web Developers . Version : Mars . 1 Release ( 4 . 5 . 1 ) Build id : 20150924 - 1200`

JDK version: `JDK 1 . 7 or later`

- **Maven project**

  emrmavensample.zip
- **Normal project**

  emrsample.zip

IntelliJ IDEA

IntelliJ IDEA version used by the reference project: `IntelliJ IDEA 15 . 0 . 1`

JDK version: `JDK 1 . 7 or later`

- **Maven project**

  emrmavensample.zip
- **Normal project**

  emrsample.zip

4 Python SDK

4.1 Install SDK

This section describes the installation process of Aliyun E-MapReduce Python 2.7+ SDK.

Install SDK

- You can install the SDK by using pip as follows.

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

- You can update the E-MapReduce Python SDK as follows.

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

- You can uninstall the SDK as follows.

```
sudo pip uninstall aliyun-python-sdk-emr
```

SDK source code

Download link

4.2 Sample code

Python sample code

Query the cluster list

```
#! /usr/bin/python
from aliyunsdkcore import client
from aliyunsdkemr.request.v20160408 import ListClusterRequest
clt = client.AcsClient('SFAW**********','Nc2nZ6dQoiqck0**********','cn-hangzhou') # set acessId and accessKey
request = ListClusterRequest.ListClusterRequest()
request.set_accept_format('xml') # xml or json
# You can set filter conditions to obtain clusters in the RUNNING or IDLE state. The state parameter is optional. You can choose not to specify this parameter.
request.add_query_param('StatusList.1', 'RUNNING')
request.add_query_param('StatusList.2', 'IDLE')
result = clt.do_action(request)
```

```
print result
```

Create a cluster

```
#! / usr / bin / env python
from aliyunsdkc ore import client
from aliyunsdke mr . request . v20160408 import CreateClus
terRequest
clt = client . AcsClient (' SFAW ***********',' Nc2nZ6dQoi qck0
**********',' cn - hangzhou ') # set acessId and accessKey
request = CreateClus terRequest . CreateClus terRequest ()
request . set_Name (" pydemo ")
request . set_ZoneId (" cn - hangzhou - b ")
request . set_LogEna ble ( False )
request . set_Securi tyGroupId (" sg -********")
request . set_IsOpen PublicIp ( True )
request . set_Charge Type (" PostPaid ")
request . set_EmrVer (" EMR - 1 . 3 . 0 ")
request . set_Cluste rType (" HADOOP ")
request . set_IoOpti mized ( True )
request . set_Instan ceGenerati on (" ecs - 2 ")
# set EcsOrder
request . add_query_ param (' EcsOrder . 1 . NodeCount ', ' 1 ')
request . add_query_ param (' EcsOrder . 1 . NodeType ', ' MASTER
')
request . add_query_ param (' EcsOrder . 1 . InstanceTy pe ', '
ecs . n1 . large ')
request . add_query_ param (' EcsOrder . 1 . DiskType ', '
CLOUD_EFFI CIENCY ')
request . add_query_ param (' EcsOrder . 1 . DiskCapaci ty ', '
80 ')
request . add_query_ param (' EcsOrder . 1 . DiskCount ', ' 1 ')
request . add_query_ param (' EcsOrder . 1 . Index ', ' 1 ')
request . add_query_ param (' EcsOrder . 2 . NodeCount ', ' 3 ')
request . add_query_ param (' EcsOrder . 2 . NodeType ', ' CORE ')
request . add_query_ param (' EcsOrder . 2 . InstanceTy pe ', '
ecs . n1 . large ')
request . add_query_ param (' EcsOrder . 2 . DiskType ', '
CLOUD_EFFI CIENCY ')
request . add_query_ param (' EcsOrder . 2 . DiskCapaci ty ', '
80 ')
request . add_query_ param (' EcsOrder . 2 . DiskCount ', ' 4 ')
request . add_query_ param (' EcsOrder . 2 . Index ', ' 2 ')
request . set_accept _format (' json ')
result = clt . do_action ( request )
print result
```

Notice:

All SDKs are generated automatically for various Alibaba Cloud services. You must modify the original code of some SDKs to apply to specific use cases. For example, in the current Python SDK, you must refer to the previous sample code when you use a List as a request parameter. If you specify a List with values of basic data types for request parameters, see configurations of the StatusList parameter in Query the list of clusters. If you specify a List with objects for request parameters, see configurations of the EcsOrder parameter in the Create a cluster section. For Python

APIs that use a List as a request parameter such as BootstrapAction, you can refer to the same places described in the preceding sentence according to a List type. We recommend that you use the Java SDK in which a List can be used properly.

For more information, see API overview.

5 SDK Reference