E-MapReduce

User Guide

User Guide

Cluster

Create a cluster

Enter the cluster creation page

Log on to Alibaba Cloud E-MapReduce Console Cluster List.

Complete RAM authorization. Refer to the Role Authorization for operating steps.

Select the region above, and the created cluster appears in the corresponding region as shown in the figure below.

Click **Create a cluster** on top right.

Cluster creation process

Note: The cluster cannot be modified after creation, except for the name. Carefully confirm the necessary configurations during creation.

Proceed with the following three steps to create a cluster:

Step 1: Configure the basic cluster information. As shown in the figure below, you need to complete the configuration of basic cluster information.

Configuration item description:

Basic information

Cluster name: The cluster name is composed of Chinese, letters, figures, "-" and "_" with the length limit of 1-64 characters.

Payment type: The payment type is consistent with that of ECS. Both Subscription and Pay-As-You-Go modes are supported. You need to select the duration if Subscription mode is selected.

Purchase duration: You can select 1 month, 2 months, 3 months, 6 months, 9 months, 1 year, 2 years or 3 years.

Operation log

Operation log: The function to save the operation log is turned on by default. In the default state, you can select the OSS directory location to save the operation log. You must activate OSS before using the function. The charge depends on the quantity of uploaded files. You are strongly recommended to open the OSS log saving function which is of great help for your in operation debugging and error screening.

Log path: OSS path to save the log.

Login setting

Login password: Set the login password at the master node. It is limited to 8-30 characters, containing both uppercase and lowercase letters as well as figures.

Bootstrap action (optional): You can execute the customized script before the cluster starts Hadoop. Refer to Bootstrap action for detailed instructions.

Step 2: Configure software. As shown in the figure below, you need to select the E-MapReduce product version and necessary software.

Configuration item description:

Product version: The main version of E-MapReduce represents a complete open source software environment and can be upgraded regularly based on the upgrade of internal component software. If the software related to Hadoop is upgraded, E-MapReduce will be upgraded, and the main version will also be upgraded. The cluster of a lower version cannot be upgraded to a higher version.

Note: Since Version E-MapReduce-2.0.0, the cluster types have no longer been distinguished clearly. All clusters are Hadoop clusters. The only difference is internal service.

Inclusion configuration: All software component lists under the selected type of cluster are presented, including name and version number. You can select different components as required. The selected component can start relevant service processes by default.

Note: The more components you select, the higher requirement for your computer configuration there will be. Otherwise there may be insufficient resources to run these services.

Software configuration (optional): Hadoop, Spark, Hive and other basic software in the cluster can be configured. Refer to Software configuration for detailed instructions.

Step 3: Configure hardware. As shown in the figure below, you need to configure the hardware.

Note: To ensure the normal use of clusters, the public network IP address is activated by default during cluster creation.

Configuration item description:

Cluster network configuration

Availability zone of the cluster: Select the availability zone where the cluster is located. The machine type and disk vary with the availability zone. There are several availability zones in each region. The availability zone belongs to different physical areas. You are generally recommended to select the same availability zone if better network connectivity is required. However, the risk of cluster creation failure will be increased as the library of single availability zone may be insufficient.

Network type: The classic network and Virtual Private Cloud (VPC) can be selected. VPC requires additional subordinate VPC and subnet (vswitch). Click Create VPC/subnet (vswitch) on the page to enter the VPC console for creation. And then click the refresh list to see the created VPC/subnet (vswitch). Refer to Virtual Private Cloud for detailed E-MapReduce VPC.

Note: The classic network is not interoperable with VPC, and the network type cannot be changed after purchase.

ECS instance series: The instance series vary with availability zones, including series I and/or series II.

New security group: Generally, there is no security group for new users. Open "New security group" and fill in the name of the security group in "Security group name".

Main security group: The security group that the cluster belongs to. Only the security group created in E-MapReduce product by the user is presented here. The security group cannot currently be created outside the E-MapReduce. To create a security group, select "New security group" and enter the name of security group. The name must be 2-64 characters and start with an upper/lowercase letter or Chinese character. Chinese, letters, figures, "-" and "_" can be used.

Cluster node configuration

High availablity cluster: After opening, Hadoop cluster has two masters to support the high availability of Resource Manager and Name Node. originally, HBase cluster supports the high availability, but a core node serves as another node. If the high availability is opened, an independent master node will be used to support it, which is safer and more reliable. The default mode is non-high availability mode, and there is only 1 master node.

E-MapReduce node configuration: In consideration of practical effects, E-MapReduce selects some types.

Data disk type: The data disks used by cluster node are ordinary cloud disk, high-efficiency cloud disk and SSD cloud disk which may vary with machine type and region. If the user selects different regions, the drop-down box will present the disks that are supported by the regions. The data disk is set to release with the cluster release by default.

Data disk volume: The recommended minimum cluster volume of a single machine is 40G and the maximum is 8000G.

Instance quantity: The quantity of instances of all required nodes. A cluster requires at least 3 instances (the high availablity cluster requires 4 instances at least, adding 1 master node). The maximum is currently 50. If more than 50 instances are required, contact us through a ticket.

Purchase list and cluster charge

The configuration list and cluster charge of your created cluster are shown on the right side of the page. The presented price information varies with the type of payment. For Subscription cluster, the total expense is shown. For Pay-As-You-Go cluster, hourly expense is shown.

Confirm creation

After all valid information is filled in, the "Create" button is active. Verify the information, and click **Create** to create the cluster.

Note:

The cluster will be created immediately if it is a Pay-As-You-Go cluster. It will go back to the Cluster List page where there is a cluster in the "Creating Cluster" status. Be patient. It will take several minutes to create a cluster. After creation, the cluster will be switched to the **Idle** status.

For Subscription cluster, the cluster will not be created before the order is generated and paid.

Creation failed

In case of creation failure, the cluster list page shows "Cluster creation failed". The reason for failure can be seen when the cursor is placed on the red exclamation point.

Expand a cluster

When your cluster resources (computing resource and storage resource) are insufficient, you can expand the cluster horizontally. Only the Core node can currently be expanded. The hardware configuration is defaulted to be consistent with that of the ECS that you purchased before.

Expansion entry

Find out the clusters to be expanded on the cluster list page. Click **Adjust scale** to enter the cluster expansion page. You can also click **View details** to enter cluster details page, and click **Adjust cluster scale** at the Core node information position on the details page.

Expansion interface

Note: Only expansion is supported now and reduction is not.

Quantity of current Master node: The quantity is 1 by default and cannot be adjusted.

Quantity of current Core: Display the quantity of all your current Core nodes by default.

Increase Core: Enter the quantity that you need to increase by. The total expense of expanded cluster is displayed on the right side. Click Confirm to start expansion. For safety, you are strongly recommended to expand small batches consisting of 1 or 2, rather than expanding a large quantity each time.

Expansion status

You can view the cluster expansion status through the Core node information on the details page. The node that is being expanded is displayed as “Expanding”. When the status of this ECS changes to “Normal”, it indicates that the ECS has been added into the cluster and can provide services normally.

Release a cluster

Click the release button at the right side of cluster entry on the cluster list page to release the cluster. Only clusters in the following statuses can be released.

Creating

Running

Idle

Ordinary release

Prompt you to confirm again before release. Once confirmed, operations are as follows:

All jobs in the cluster will be forcibly terminated.

If you have selected to save the log to OSS, all current job logs will be saved to OSS. The required time depends on the log size. The log uploading and the job running are in parallel. The log can be uploaded once generated. Therefore, when the final job is terminated, few logs need to be uploaded. They may be uploaded within several minutes.

Release a cluster.

Forcible release

If you don’ t need any logs but want to terminate the cluster operation as soon as possible, activate the forcible release function. The process of log collection will be skipped (if the log collection is active) and release will start directly.

Cluster release failure

Due to system error or other causes, the cluster release may fail after confirmation. Your clicking and the release are asynchronous, so the cluster release may fail after a while. But don’ t worry, E-MapReduce can start background protection to automatically release the cluster again until it is released successfully.

Cluster list page

Display the basic information of all your clusters.

All items in the list column are described as follows:

Instance ID/cluster name: ID and name of the cluster. Place the cursor on the name to modify the cluster name.

Cluster type: There is only Hadoop type available now.

Elapsed time: The operating time from creation until now. Once the cluster is released, the time stops.

Status (default): The status of cluster. Please refer to **Cluster status**. In case of cluster abnormalities such as creation failure, the prompt information will appear on the right side. The detailed error information can be viewed by hovering the mouse. You can also filter the status by clicking **Status (default)**.

Payment type: The payment type of the cluster.

Operation: The operation that can be applied to the current cluster, including the following content.

View details: Enter the cluster details page and view the detailed information after cluster creation.

Adjust scale: The entry for cluster expansion function.

Release: Release a cluster. Please refer to **Release cluster**.

Cluster details page

Displays the details of clusters, including the following four parts:

Cluster information

ID/name: ID and name of cluster instances.

Payment type: The payment type of the cluster.

Region: The region where the cluster is located.

Current status: Refer to **Cluster status**.

Start time: The creation time of the cluster.

Running time: The running time of the cluster.

Log activation: Whether the log saving function is activated.

Log position: The position where the log is saved is displayed here if the log saving function is activated.

Software configuration: Information about software configuration.

Bootstrap/software configuration: Whether there are abnormities.

High availability cluster: Whether the high availability cluster is activated.

Bootstrap action

The names, paths and parameters of all configured bootstrap actions are listed here.

Software information

Main version: Use the main version of E-MapReduce.

Cluster type: The selected cluster type.

Software information: All application programs installed by the user and their versions are listed here, such as Hadoop 2.6.0 and Hive 0.14.

Hardware information

Network type: Network type which the cluster is using.

Main security group: The security group the cluster belongs to.

Main availability zone: The availability zone where the cluster is located, for example, cn-hangzhou-b. It is consistent with that of ECS.

VPC/subnet: VPC where the user cluster is located and ID of subnet VSwtich.

Master node information: Configurations corresponding to all Master nodes, including the following content.

Quantity of nodes: Quantity of current nodes and actually applied nodes. The two values are the same theoretically. However, during creation, the quantity of current nodes is less than that of applied nodes until the creation is complete.

CPU: The quantity of CPU cores at a single node.

Memory: The volume of memory at a single node.

Data disk type: The type of data disk.

Data disk: The volume of data disk at a single node.

Instance status: Including creating, normal, expanding and released.

Public network IP: The public network IP address of Master.

Intranet IP: The Intranet IP address of a machine which can be visited by all nodes in the cluster.

ECS expiration time: The expiration time for use of ECS purchased.

E-MapReduce expiration time: the expiration time for use of E-MapReduce purchased.

Core node information: Configurations corresponding to all Core nodes, including the following content.

Quantity of nodes: Quantity of current nodes and actually applied nodes.

CPU: The quantity of CPU cores at a single node.

Memory: The volume of memory at a single node.

Data disk type: The type of data disk.

Data disk: The volume of hard disk at a single node.

Instance status: Including creating, normal and expanding.

Intranet IP: The Intranet IP address of a machine which can be visited by all nodes in the cluster.

ECS expiration time: The expiration time for use of ECS purchased.

E-MapReduce expiration time: The expiration time for use of E-MapReduce purchased.

Cluster script

When a cluster, especially a yearly or monthly cluster, is used, new third-party software may be

installed, resulting in the need to change the cluster runtime environment. After a cluster is created, the cluster script allows you to select nodes in batches and run your specified script in order to fulfill your individual requirements.

A cluster script is similar to a bootstrap action. After creating a cluster, you can install many software packages, which are not currently supported by E-MapReduce, to your cluster. For example:

Use Yum to install the software that has been provided.

Directly download some public software packages from the public network.

Read your own data from OSS.

Install and run a service like Flink or Impala, but the script to be compiled will be more complex.

It is strongly recommended that you test the cluster script first on a single node, and perform operations on the whole cluster after the cluster script is tested to be correct.

How to use a cluster script

A cluster script can run on an idle or running cluster. On the cluster list page, click the **Display details** button of corresponding cluster.

Click **Cluster script** on the left-side menu to enter the cluster script execution interface. On the right side is a list of cluster scripts that have been executed.

Click **Create and execute** at the top right corner to enter the creation interface.

Fill in the creation interface with configuration items. Select the node for execution and click “Execute” to confirm and execute the operations.

You can see the newly created cluster script in the cluster script list and click “Refresh” to update the cluster script status.

You can click **Display details** to display the running status of a script on each node, or click “Refresh” to update the running status of a cluster on each node.

The cluster script can run only on available clusters that are idle or running. The cluster script is applicable for long-standing clusters. For temporary clusters created on demand, perform a bootstrap

action to initialize the clusters.

The cluster script will download a script from the OSS and run it on your specified node. When the returned value is 0, the execution fails. If the execution fails, you can log on to each node to check the running log. The running log is located at /var/log/cluster-scripts/clusterScriptId of each node. If the cluster has been configured with an OSS log directory, the running log will also be uploaded to osslogpath/clusterId/ip/cluster-scripts/clusterScriptId for easy viewing.

By default, the root account will be used to execute your specified script. In the script, you can use “su hadoop” to switch to Hadoop account.

A cluster script may be successfully executed on some nodes but fail on other nodes. For example, restart of a node can lead to a failure in script operation. After resolving the error, you can execute the cluster script again on the failed node. After a cluster is expanded, you can also specify the expanded node for separate execution of the cluster script.

Only one cluster script can run on a cluster each time. If a cluster script is running, you cannot submit a new cluster script for execution. For each cluster, you can retain at most 10 cluster script records. If you need to retain more than 10 script records, you must delete the previous records first.

Script example

For a script similar to a bootstrap action, you can specify the file in the script to be downloaded from OSS. In the following example, the file oss://yourbucket/myfile.tar.gz is downloaded and decompressed to the directory /yourdir:

```
#!/bin/bash
osscmd --id=<yourid> --key=<yourkey> --host=oss-cn-hangzhou-internal.aliyuncs.com get
oss://<yourbucket>/<myfile>.tar.gz ./<myfile>.tar.gz
mkdir -p /<yourdir>
tar -zxvf <myfile>.tar.gz -C /<yourdir>
```

OSSCMD is pre-installed on the node and can be called directly to download the file.

NOTE: The OSS host address can be an Intranet address, an Internet address, or a VPC network address. If a classic network is used, you need to specify an Intranet address. If the network is located in Hangzhou, the Intranet address is oss-cn-hangzhou-internal.aliyuncs.com. If a VPC network is used, you need to specify a domain name that can be accessed from the VPC Intranet. If the network is located in Hangzhou, the domain name is vpc100-oss-cn-hangzhou.aliyuncs.com.

Additional system software packages can be installed to the script through Yum, for example, ld-linux.so.2：

```
#!/bin/bash
yum install -y ld-linux.so.2
```

Cluster renewal

When your Subscription cluster service becomes due soon, you need to renew the cluster to continue your E-MapReduce cluster services. The cluster renewal includes the renewal of E-MapReduce service charge and ECS fees.

Renewal entrance

Log on to Alibaba Cloud E-MapReduce Cluster List Page.

Determine the cluster to be renewed.

Click **Renewal** under the corresponding cluster entry operation column to enter the cluster renewal page.

Renewal page

Renewal: Tick the machine to be renewed.

ECS instance ID: ECS instance ID of the machine in the cluster.

Current ECS expiration time: The expiration time of the ECS.

Current E-MapReduce expiration time: The expiration time of E-MapReduce product.

ECS renewal time: The duration of the renewal. Currently, 1-9 months, 1 year, 2 years and 3 years are supported.

E-MapReduce renewal time: The duration of E-MapReduce service fee at the node which is recommended to be consistent with that of ECS.

ECS renewal price: Corresponding renewal price of the ECS nodes.

E-MapReduce renewal price: The renewal price of corresponding nodes of the E-MapReduce service.

Pay for the order

Note: The fees of cluster renewal is the sum of ECS renewal price and E-MapReduce service product price. If there are unpaid orders in the cluster list, you cannot expand or renew any cluster.

Click **Confirm** to view the prompt box for successful order placement (be patient as the prompt information may be delayed for a long time).

Click **Pay for the order** to skip to the order payment page. The payment page displays the total amount payable and order details. One of them is the order of E-MapReduce product fees, and others are the ECS orders of cluster renewal.

Click **Confirm payment** to complete the payment.

After payment, click **Completed** to return to the cluster list page.

At this time, the expiration time of successfully renewed clusters displayed on the cluster list page will be updated to the time after renewal. The expiration time of corresponding ECS will usually be updated to the time after renewal 3 to 5 minutes later.

If you confirm the renewal order but never pay for it, you can find the cluster entry on the cluster list page. **Pay** and **Cancel** are displayed in the operation column on the right side. You can click **Pay** to complete the corresponding order payment and cluster expansion processes, or click **Cancel** to cancel the renewal.

Security group

The security group that is created in E-MapReduce will be used during cluster creation.

The main reason is that only port 22 is opened in the cluster created by E-MapReduce. You are recommended to divide the ECS instance by function and place it into different user security groups. For example, the security group of E-MapReduce is “E-MapReduce security group”, while the security group that you have already created is “User-security group”. Each security group is provided with unique access control as required.

If it is necessary to link with the cluster that has already been created, refer to the following method.

Add E-MapReduce cluster to existing security group

Log on to Alibaba Cloud E-MapReduce Console Cluster List.

Find the cluster entry to be added to the security group. Click **View details** to enter the cluster details page.

Find the security group under “Network information” on the cluster details page. View the name and ID of the security group where all ECS instances are located.

Enter Alibaba Cloud ECS Management Console. Click **Security group** on the left side of the page to find the security group entry in the list as viewed in Step 3.

Click **Manage instances** in the security group, and then you can see many ECS instance names starting with emr-xxx. These are corresponding ECS in the E-MapReduce cluster.

Select all these instances and click **Move to security group**, and then select the new security group to add into.

Add the existing cluster into E-MapReduce security group

Find the security group where the existing cluster is located, repeat abovementioned operations and move to E-MapReduce security group. Select scattered machines on the ECS console directly, and then move the clusters to E-MapReduce security group in batch.

Security group rules

The security group rules are subject to the “or” relationship when an ECS instance is in several different security groups. For example, only port 22 of the E-MapReduce security group is opened, while all ports of “User - security group” are opened. After the cluster of E-MapReduce is added into “User - security group”, all ports of the machine in E-MapReduce will be opened. Pay special attentions during use.

Quick portal for component viewing

When a cluster is created, several domain names are bound to the cluster by default for you to access your open source components:

- Yarn
- HDFS
- Ganglia

These links can be found in **Quick portal for component** in **Cluster management**.

By default, there is no username and password for accessing. Therefore, the access request cannot pass the HTTP authentication. You need to click **Access setting** to set an access username and password to access your component UI interface.

Only one username and one password can be used. Therefore, the new username and password will always replace the previous ones.

NOTE: Currently, this function is only supported by Version 2.3 and above.

Job

Hadoop MapReduce job configuration

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select Hadoop job type to create a Hadoop Mapreduce job. This type of job is Hadoop job submitted in the background via the following process.

```
hadoop jar xxx.jar [MainClass] -Dxxx ....
```

Fill in the **Parameters** with command line parameters required to submit this job. Note that the content to be filled in this option box shall be started with the first parameter after "hadoop jar". That is to say, in the option box, the address of the jar package required to run this job is the first to be filled in with, followed by [MainClass] and other command line parameters you can provide on your own.

For example, if you want to submit a Hadoop sleep job which doesn't write/read any data, this job will succeed just by submitting mapper reducer tasks to the cluster and waiting for each task to sleep for a while. In Hadoop (such as hadoop-2.6.0), this job is packaged in hadoop-mapreduce-client-jobclient-2.6.0-tests.jar of the Hadoop release version. If this job is submitted from the command line, the command will be:

```
hadoop jar /path/to/hadoop-mapreduce-client-jobclient-2.6.0-tests.jar sleep -m 3 -r 3 -mt 100 -rt 100
```

To configure this job in E-MapReduce, the content to be filled in the option box of **Parameters** on the configuration page will be:

```
/path/to/hadoop-mapreduce-client-jobclient-2.6.0-tests.jar sleep -m 3 -r 3 -mt 100 -rt 100
```

Note that since the path for the jar package here is an absolute path of E-MapReduce host machine, there may be a problem - the user may store these jar packages under any path and these jar packages will be released and not available as the cluster is created and released. Therefore, use the following methods:

You shall upload your jar packages to the OSS bucket for storage; when configuring Hadoop parameters, click **Select OSS path** to select jar packages to be performed from the OSS directory. The system will complete the OSS address for jar packages automatically. Switch the jar prefix of the code to "ossref" (click **Switch resource type**) to guarantee this jar package can be downloaded correctly by E-MapReduce.

Click **OK** to automatically fill in the option box of **Parameters** with OSS path of this package. When a job is submitted, the system will find the corresponding jar packages automatically as per this path.

You can further complete other command line parameters for job running behind the jar package path of this OSS.

Select the policy for failed operations.

Click **OK** to complete job configuration definition.

In the above example, sleep job has no data input/output. If the job needs to read data and process input results (e.g. wordcount), the data input and output paths are required to be specified. You can read/write the data on HDFS of E-MapReduce cluster as well as on OSS. To read/write the data on OSS, just write the data path as the OSS path when filling in input and output paths. For example:

```shelljar ossref://emr/checklist/jars/chengtao/hadoop/hadoop-mapreduce-examples-2.6.0.jar randomtextwriter -D mapreduce.randomtextwriter.totalbytes=320000 oss://emr/checklist/data/chengtao/hadoop/Wordcount/Input
```

Hive Job Configuration

In E-MapReduce, when users are applying for a cluster, they are provided with Hive environment by default and can directly create and perform their table and data by using Hive. Operation steps are as follows.

You need to prepare the Hive script in advance, for example:

```
 USE DEFAULT;
set hive.input.format=org.apache.hadoop.hive.ql.io.HiveInputFormat;
set mapreduce.job.maps=12;
set mapreduce.job.reduces=6;
set hive.stats.autogather=false;

DROP TABLE uservisits;

CREATE EXTERNAL TABLE IF NOT EXISTS uservisits (sourceIP STRING,destURL STRING,visitDate
STRING,adRevenue DOUBLE,user
Agent STRING,countryCode STRING,languageCode STRING,searchWord STRING,duration INT ) ROW
FORMAT DELIMITED FIELDS TERMI
NATED BY ',' STORED AS SEQUENCEFILE LOCATION '/HiBench/Aggregation/Input/uservisits';

DROP TABLE uservisits_aggre;

CREATE EXTERNAL TABLE IF NOT EXISTS uservisits_aggre ( sourceIP STRING, sumAdRevenue DOUBLE)
STORED AS SEQUENCEFILE LO
CATION '/HiBench/Aggregation/Output/uservisits_aggre';
INSERT OVERWRITE TABLE uservisits_aggre SELECT sourceIP, SUM(adRevenue) FROM uservisits GROUP BY
sourceIP;
```

Save this script into a script file, such as “uservisits_aggre_hdfs.hive”, and then upload the script to an OSS directory (for example: oss://path/to/uservisits_aggre_hdfs.hive).

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Hive job type to create a Hive job. This type of job is submitted in the background via the following process.

```
hive [user provided parameters]
```

Fill in the **Parameters** option box with parameters subsequent to Hive commands. For example, if it is necessary to use Hive script just uploaded to OSS, the following shall be filled in:

```
-f ossref://path/to/uservisits_aggre_hdfs.hive
```

You can also click **Select OSS path** to view and select from OSS, and the system will complete the absolute path of Hive script on OSS automatically. Switch the Hive script prefix to "ossref" (click **Switch resource type**) to guarantee E-MapReduce can download this file correctly.

Select the policy for failed operations.

Click **OK** to complete the Hive job definition.

Pig job configuration

In E-MapReduce, when you are applying for clusters, you are provided with Pig environment by default and can directly create and perform your table and data by using Pig. Operation steps are as follows.

You need to prepare Pig script in advance, for example:

```
 ```shell
/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

-- Query Phrase Popularity (Hadoop cluster)
```
```

```
-- This script processes a search query log file from the Excite search engine and finds search phrases that
occur with particular high frequency during certain times of the day.

-- Register the tutorial JAR file so that the included UDFs can be called in the script.
REGISTER oss://emr/checklist/jars/chengtao/pig/tutorial.jar;

-- Use the PigStorage function to load the excite log file into the “raw” bag as an array of records.
-- Input: (user,time,query)
raw = LOAD 'oss://emr/checklist/data/chengtao/pig/excite.log.bz2' USING PigStorage('\t') AS (user, time,
query);

-- Call the NonURLDetector UDF to remove records if the query field is empty or a URL.
clean1 = FILTER raw BY org.apache.pig.tutorial.NonURLDetector(query);

-- Call the ToLower UDF to change the query field to lowercase.
clean2 = FOREACH clean1 GENERATE user, time, org.apache.pig.tutorial.ToLower(query) as query;

-- Because the log file only contains queries for a single day, we are only interested in the hour.
-- The excite query log timestamp format is YYMMDDHHMMSS.
-- Call the ExtractHour UDF to extract the hour (HH) from the time field.
houred = FOREACH clean2 GENERATE user, org.apache.pig.tutorial.ExtractHour(time) as hour, query;

-- Call the NGramGenerator UDF to compose the n-grams of the query.
ngramed1 = FOREACH houred GENERATE user, hour,
flatten(org.apache.pig.tutorial.NGramGenerator(query)) as ngram;

-- Use the DISTINCT command to get the unique n-grams for all records.
ngramed2 = DISTINCT ngramed1;

-- Use the GROUP command to group records by n-gram and hour.
hour_frequency1 = GROUP ngramed2 BY (ngram, hour);

-- Use the COUNT function to get the count (occurrences) of each n-gram.
hour_frequency2 = FOREACH hour_frequency1 GENERATE flatten($0), COUNT($1) as count;

-- Use the GROUP command to group records by n-gram only.
-- Each group now corresponds to a distinct n-gram and has the count for each hour.
uniq_frequency1 = GROUP hour_frequency2 BY group::ngram;

-- For each group, identify the hour in which this n-gram is used with a particularly high frequency.
-- Call the ScoreGenerator UDF to calculate a "popularity" score for the n-gram.
uniq_frequency2 = FOREACH uniq_frequency1 GENERATE flatten($0),
flatten(org.apache.pig.tutorial.ScoreGenerator($1));

-- Use the FOREACH-GENERATE command to assign names to the fields.
uniq_frequency3 = FOREACH uniq_frequency2 GENERATE $1 as hour, $0 as ngram, $2 as score, $3 as
count, $4 as mean;

-- Use the FILTER command to move all records with a score less than or equal to 2.0.
filtered_uniq_frequency = FILTER uniq_frequency3 BY score > 2.0;

-- Use the ORDER command to sort the remaining records by hour and score.
ordered_uniq_frequency = ORDER filtered_uniq_frequency BY hour, score;

-- Use the PigStorage function to store the results.
-- Output: (hour, n-gram, score, count, average_counts_among_all_hours)
```

```
STORE ordered_uniq_frequency INTO 'oss://emr/checklist/data/chengtao/pig/script1-hadoop-results'
USING PigStorage();
```

Save this script into a script file,such as “script1-hadoop-oss.pig”, and then upload the script to an OSS directory (for example: oss://path/to/script1-hadoop-oss.pig).

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Pig job type to create a Pig job. This type of job is submitted in the background via the following process.

```
pig [user provided parameters]
```

Fill in the option box of **Parameters** with parameters subsequent to Pig commands. For example, if it is necessary to use a Pig script just uploaded to OSS, the following shall be filled in:

```
-x mapreduce ossref://emr/checklist/jars/chengtao/pig/script1-hadoop-oss.pig
```

You can click **Select OSS path** to view and select from OSS, and the system will complete the absolute path of Pig script on OSS automatically. Switch Pig script prefix to “ossref” (click **Switch resource type**) to guarantee E-MapReduce can download this file correctly.

Select the policy for failed operations.

Click **OK** to complete the Pig job definition.

Spark job configuration

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Spark job type to create a Spark job. In E-MapReduce back-end, Spark jobs are submitted via the following process:

```
spark-submit [options] --class [MainClass] xxx.jar args
```

Fill in the option box of **Parameters** with command line parameters required to submit this Spark job. Note that the box shall be filled in with parameters after "spark-submit" only. Examples below show how to create parameters for Spark jobs and pyspark jobs respectively.

Create a Spark job

Create a Spark WordCount job.

Job name: Wordcount

Type: Select Spark

Parameters:

Complete command submitted to the command line:

```
spark-submit --master yarn-client --driver-memory 7G --executor-memory 5G --executor-cores 1 --num-executors 32 --class com.aliyun.emr.checklist.benchmark.SparkWordCount emr-checklist_2.10-0.1.0.jar oss://emr/checklist/data/wc oss://emr/checklist/data/wc-counts 32
```

Only fill in the parameter box of E-MapReduce job with:

```
--master yarn-client --driver-memory 7G --executor-memory 5G --executor-cores 1 --num-executors 32 --class com.aliyun.emr.checklist.benchmark.SparkWordCount ossref://emr/checklist/jars/emr-checklist_2.10-0.1.0.jar oss://emr/checklist/data/wc oss://emr/checklist/data/wc-counts 32
```

Note that job Jar packages are saved in OSS and the method to refer to this Jar package is ossref://emr/checklist/jars/emr-checklist_2.10-0.1.0.jar. You can click **Select OSS path** to view and select from OSS, and the system will complete the absolute path of Spark script on OSS

automatically. Switch the default “oss” protocol into “ossref” protocol.

Create a pyspark job

In addition to jobs of Scala or Java types, E-MapReduce also supports Spark jobs of python type. Create a Spark Kmeans job for python script:

Job name: Python-Kmeans

Type: Spark

Parameters:

```
--master yarn-client --driver-memory 7g --num-executors 10 --executor-memory 5g
--executor-cores 1 --jars ossref://emr/checklist/jars/emr-core-0.1.0.jar
ossref://emr/checklist/python/wordcount.py oss://emr/checklist/data/kddb 5 32
```

References of Python script resource are supported and “ossref” protocol is used.

For pyspark, online Python installation kit is not currently supported.

Select the policy for failed operations.

Click **OK** to complete the Spark job definition.

Spark SQL job configuration

Note that the mode of Spark SQL to submit job is yarn-client mode by default.

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Spark SQL job type to create a Spark SQL job. In E-MapReduce back-end, Spark SQL job is submitted via the following process:

```
spark-sql [options] [cli option]
```

Fill in the option box of **Parameters** with parameters subsequent to Spark SQL commands.

"-e" option

Directly write running SQL for "-e" options by inputting it into the **Parameters** box of the job directly as shown below:

```
-e "show databases;"
```

"-f" option

"-f" options can be used to specify Spark SQL script file. Loading well prepared Spark SQL script files on OSS can obtain more flexibility, and you are recommended to use to this operation mode. As shown below:

```
-f ossref://your-bucket/your-spark-sql-script.sql
```

Select the policy for failed operations.

Click **OK** to complete Spark SQL job definition.

Shell job configuration

Note that Shell script is currently executed by users using Hadoop by default. If it is required to use root user, sudo can be used. Shell script must be used with caution.

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Shell job type to create a Bash Shell job.

Fill in the option box of **Parameters** with parameters subsequent to Shell commands.

"-c" option

"-c" option can be used to directly set Shell scripts to run by filling in the **Parameters** box of the job directly, as shown below:

```
-c "echo 1; sleep 2; echo 2; sleep 4; echo 3; sleep 8; echo 4; sleep 16; echo 5; sleep 32; echo 6; sleep 64; echo 8; sleep 128; echo finished"
```

"-f" option

"-f" option can be used to directly run Shell script files. By uploading a Shell script file to OSS, Shell scripts on OSS will be directly formulated in the job parameters. This is more flexible than that of "-c" option, as shown below:

```
-f ossref://mxbucket/sample/sample-shell-job.sh
```

Select the policy for failed operations.

Click **OK** to complete Shell job definition.

Sqoop job configuration

Note that only E-MapReduce products version V1.3.0 (included) and higher support Sqoop job type. Running Sqoop job on low version clusters will fail, and errlog will report errors "Not supported". Refer to Data Transmission Sqoop for parameter details.

Log on to Alibaba Cloud E-MapReduce Console Job List.

Click **Create a job** on the top right corner of this page to enter the job creating page.

Fill in the job name.

Select the Sqoop job type to create a Sqoop job. In E-MapReduce back-end, Sqoop job is to submit via the following process:

```
sqoop [args]
```

Fill in the option box of **Parameters** with parameters subsequent to Sqoop commands.

Select the policy for failed operations.

Click **OK** to complete Sqoop job definition.

Job operations

Job creation

A new job can be created at any time. The job created can currently only be used in the region where it is created.

Job clone

To completely clone configurations in which jobs already exist. It is also restricted to the same region.

Job modification

If it is necessary to add a job into an execution plan, then it is required to ensure that this execution plan is not under operation and its periodic scheduling is not in progress before the job can be modified.

If it is necessary to add this job into several execution plans, the modification can be made after terminating the operation and periodic scheduling of all execution plans to be added. Since job modification may cause changes to all execution plans using this job, it may cause errors to execution plans under executing or periodic scheduling.

If it is necessary to conduct debugging, clone is recommended, and after debugging, replace original jobs in the execution plan.

Job deletion

As with modification, jobs can be deleted only when the execution plan in which jobs are added in is not under operation and periodic scheduling is not in progress.

Job date variables

During creation, time variable wildcard settings in job parameter are supported.

Variable wildcard format

The format of variable wildcard supported by E-MapReduce is ${dateexpr-1d} or ${dateexpr-1h}. For example, assuming the current time is “20160427 12:08:01”:

> If it is written as ${yyyyMMdd HH:mm:ss-1d} in job parameters, then this parameter wildcard will be, when executed practically, replaced with “20160426 12:08:01”, that is, the current date minus one day with accuracy to the second.
>
> If it is written as ${yyyyMMdd-1d}, then it will be replaced with “20160426” when being executed, representing the day before current date.
>
> If it is written as ${yyyyMMdd}, then it will be replaced with “20160427”, representing the current date.

The dateexpr represents the expression of standard time format, and corresponding time will be formatted as per this expression and followed by corresponding time to add or deduct. Following the expression, 1d (1 day) to add or deduct can be written as N days or hours, for example, ${yyyyMMdd-5d}, ${yyyyMMdd+5d}, ${yyyyMMdd+5h} and ${yyyyMMdd-5h} are all supported, and corresponding replacing methods are consistent with the descriptions above.

Note: E-MapReduce currently supports addition and deduction only for “hour” and “day”, that is, the format of +Nd, -Nd, +Nh and -Nh after dateexpr (dateexpr refers to the expression of time format and N is an integer).

Example

When being executed practically, the **Parameter** in the job on the figure below will be replaced with:

```
jar ossref://emr/jar/hadoop/hadoop_wc.jar com.aliyun.emr.WordCount  oss://emr/output/pt=20160426
```

Execution plan

Create an execution plan

An execution plan is a set of jobs that can be executed one-time or periodically through scheduling configuration. It can be executed on an existing E-MapReduce cluster and also can create a temporary cluster as required to execute assignment dynamically. Its biggest advantage is to use resources actually needed during execution so as to maximize resource savings.

The steps to create an execution plan are as follows:

Log on to Alibaba Cloud E-MapReduce Console Plan Execution Page.

Select the region.

Click **Create an execution plan** at the top right corner, and enter the **Create Execution Plan** page.

There are two options on the **Select Cluster Mode** page, **Create on demand** and **Existing clusters**.

Create as required: create a new cluster to run jobs.

Execution plan of one-time scheduling: clusters with corresponding configuration will be created when execution starts and then released upon the completion of operation. Refer to Create a Cluster for specific descriptions of creation parameters.

Execution plan of periodic scheduling: a new cluster will be created as per users' settings when each scheduling period starts and then released upon the completion of operation.

Existing clusters: use an existing cluster which complies with the following requirements. If selecting **Existing clusters**, then enter the **Select Cluster** page. You can select the cluster with which the execution plan will be associated.

i. Currently, execution plan can only be submitted to clusters in **Running** or **Idle** status.

Click **Next** to enter the job configuration page. All user jobs will be listed in the left table, and you can click a job to select the job for execution. Then by clicking the right-facing

button, the checked jobs will be added into the job queue. Jobs in the checked jobs queue will be submitted to the cluster for execution as per their orders. The same job can be added and then executed several times. If you have not created any jobs, refer to operating instructions to create jobs.

Click **Next** to enter scheduling mode configuration page. The configuration items are described as follows:

Execution plan name: limited to 1-64 characters, and only Chinese, letters, numbers, “- ” and “_” are allowed.

Scheduling strategy

Manual execution: the execution plan will not be executed automatically after creation. It will be executed manually. Once the execution is in progress, it cannot be conducted again.

Periodic scheduling: this function will be enabled immediately after the execution plan is created. And the execution can be started from the scheduling time point configured. The periodic scheduling can be disabled in the list page. When the scheduling execution starts, if the execution in the last period has not been completed, this scheduling will be ignored.

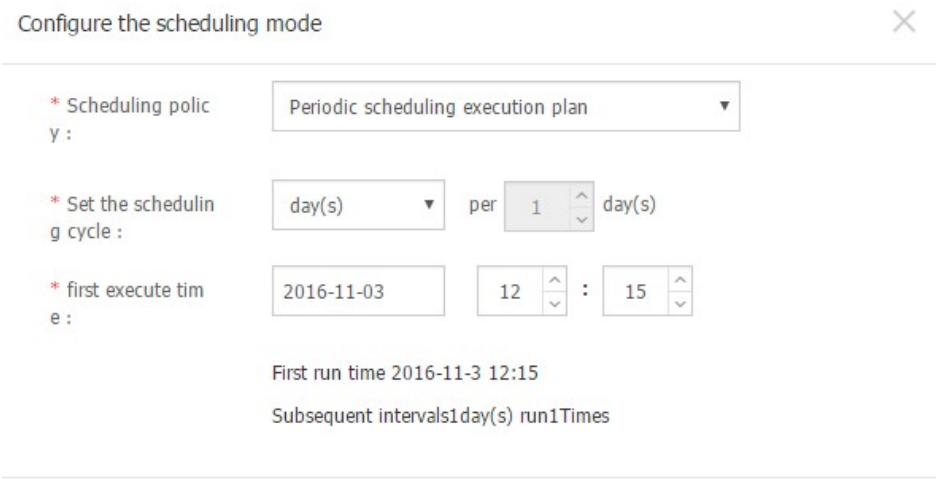
Scheduling period setting: there may be two scheduling periods - days and hours. For days, it is one day by default and cannot be changed. If the hours period is selected, the specific time interval can be set and the range is from 1 to 23.

First execution time: the effective start time of scheduling. From this time on, periodic scheduling will be conducted as per scheduling periods. The first scheduling will be conducted from the latest time point when requirements are met as per actual time.

Click **Confirm to submit** to complete the creation of the execution plan.

Others

Example for periodic scheduling



This setting indicates that the scheduling is initially started from 2016-11-03 12:15 and then conducted every other day. The second scheduling is then conducted on 2016-11-04 12:15.

Execution sequence of jobs

For jobs in the execution plan, they shall be executed from the first to the last as per the sequence of user-selected job in the Job List.

Execution sequence of multiple execution plans

Every execution plan can be deemed as an integral whole. When multiple execution plans are submitted to the same cluster, each execution plan will submit jobs in its internal job sequence, which is consistent with the sequence of a single execution plan. While jobs among multiple execution plans are in parallel.

Practical example - early job debugging

During job debugging, if the cluster is created on demand automatically, the speed will be slow and it will take a lot of time to start the cluster. The recommended way is to manually create a cluster first, and then select “Associate the cluster” in the execution plan to run jobs and to set the scheduling mode as **Execute immediately**. During debugging, results are viewed by clicking **Run** on the execution plan list page to run for multiple times. Modify execution plan once job debugging is completed, and then modify the mode of associating existing clusters into creating a new cluster on demand. And modify the scheduling mode into periodic scheduling (as appropriate). Thus, tasks will be executed automatically on demand.

Manage execution plans

Log on to Alibaba Cloud E-MapReduce Console Plan Execution Page.

Find corresponding execution plan items, and click the **Manage** button in its **Action Bar** to enter the execution plan management page. Here you can:

View details of the execution plan

You can view basic information such as names, associated clusters and job configurations of the execution plan as well as scheduling strategy and status, alarm information and so on.

Modify the execution plan

Considerations

Only jobs neither in the process of running nor being scheduled can be modified. For an execution plan to be executed immediately, it can be modified only when it is not running currently. If the execution plan is periodically scheduled, wait for the completion of its current operation, verify whether it is in periodical scheduling, and click **Stop scheduling** (if yes) before you modify it.

Modify independently

Every separate module can be modified independently. Click the **Modify** button on the right side of the item to perform modification.

Configure alarm notification

There are three types of alarm notifications in total:

Notification for booting timeout: if a periodical scheduling has not been conducted properly at the specified time point and is not executed within 10 minutes of timeout, an alarm will be sent.

Notification for failed execution: if any job in the execution plan failed, an alarm will be sent.

Notification for successful execution: if all jobs in the execution plan are executed successfully, a notification will be sent.

Run and view results

When the execution plan can be run, there will be a button named **Run** on the right side of **Scheduling status** in **Basic Information**. A schedule will be executed by clicking this button.

At the bottom of the page, there are running records displaying the execution plan instances executed each time, facilitating views of the corresponding job list and logs.

Execution plan list

Displays basic information of all your execution plans.

Execution plan ID/name: The ID and corresponding name of the execution plan.

Recent Execution Cluster: The last cluster to execute this execution plan. It is a cluster created on demand or an existing associated cluster. If a cluster is created automatically on demand, (Automatically created) will be displayed below the cluster, indicating that the cluster is created automatically by E-MapReduce on demand and will be released automatically after running.

Last running condition: The running status of the last execution plan.

Start time: The time from which the last plan starts to run.

Running time: The duration for which the last plan is running.

Running status: The running status of the last execution plan.

Scheduling status: Whether the scheduling is in progress or has been stopped. Only periodic jobs have the scheduling status.

Operation

Run now: Manual run can be made only when the job is neither running nor being scheduled. The execution plan will be run once immediately after clicking.

Enable/Stop scheduling: When the scheduling is stopped, **Enable scheduling** will appear which can be clicked to start scheduling. When **Stop scheduling** is displayed during scheduling, clicking it will stop the scheduling. This button is only for periodic execution plans.

Running records: Click to enter the job log viewing page in the execution plan.

More

Modify: To modify the configuration of an execution plan. The execution plan in the process of scheduling and running cannot be modified.

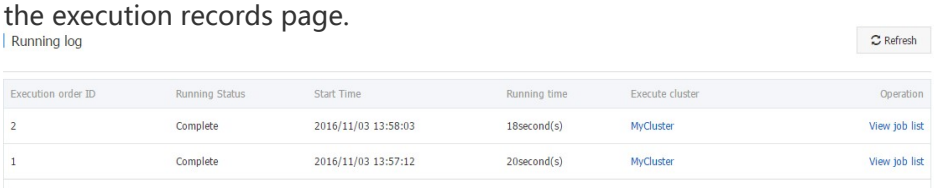
Delete: To delete an execution plan. The execution plan in the process of scheduling or running cannot be deleted.

View job results and logs

View execution records

Log on to Alibaba Cloud E-MapReduce Console Plan Execution Page.

Click the **Running records** from the right-side operation items in the execution plan to enter the execution records page.



Execution sequence ID: The times of execution of this execution record, indicating its ordinal position in the whole execution queue. For example, 1 stands for the initial time of execution while n stands for the n-th time.

Running status: The running status of each execution record.

Start time: The time on which the execution plan starts to run.

Running time: The total running time until the page is viewed.

Execution cluster: The cluster run by the execution plan, which can be either a cluster on demand or an existing associated cluster. Click to view the page of cluster details.

Operation

- View the job list: Click this button to enter the job list of a single execution plan to view the execution condition of each job.

Job record view

Here you can view the job list in execution records of a single execution plan and specific details of each job.

Job execution ID: A corresponding ID will be created every time when a job is executed, and this ID is different from the job ID. This ID can be considered as the unique identifier of the record for a job to run each time, with which you can view the logs on OSS.

Job name: The name of the job.

Status: The running status of the job.

Job type: The type of the job.

Start time: The time on which this job starts to run; it has been converted into local time.

Running time: The total running time (in seconds) of this job.

Operation

Stop the job: A job can be stopped whether it is in the process of submission or running. If a job is in submission, stopping it will cancel execution. If the job is running, it will be killed.

stdout: To record all output content from standard output (that is, Channel 1) of the master process. If log saving of the cluster which runs jobs is not enabled, this viewing function cannot be executed.

stderr: To record all output content from diagnostic output (that is, Channel 2) of the master process. If log saving of the cluster which runs jobs is not enabled, this viewing function cannot be executed.

View the job instance: To view the log of all job worker nodes. If log saving of the cluster which runs jobs is not enabled, this viewing function cannot be executed.

Job worker log view

Here you can view the log of all job worker nodes.

Elastic Compute Service (ECS) instance ID/IP: The ECS instance ID of a running job and the corresponding Intranet IP address.

Container ID: The container ID of Yarn running.

Type: Different log types. **stdout** and **stderr** come from different output.

Operation

- View the log: Click different types to view the corresponding logs.

Parallel execution of multiple execution plans

To maximize the use of available computing resources of a cluster, multiple execution plans can currently be mounted to the same cluster to utilize parallel execution.

The main points are summarized as follows:

Jobs in the same execution plan will be executed in series, and it is considered by default that subsequent jobs can be submitted and executed only after execution of preceding jobs is completed.

In case of sufficient cluster resources, it is required to create a number of different execution plans and to associate them to the same cluster for submission and running to execute multiple jobs in parallel (a cluster is considered by default to support at most 20 execution plans for parallel execution).

The management and control system currently supports parallel submission for execution plans associated to the same cluster to Yarn. However, if the cluster itself has insufficient resources, jobs will be congested in the Yarn queue to wait for scheduling.

For the process of creating execution plans and associating it to the cluster, refer to Execution Plan Creating Process.

Software Configuration

Purpose of software configuration

Hadoop, Hive, Pig and other software contain numerous configurations. The software configuration function can be used to change the software configuration. For example, the number of service threads in HDFS server dfs.namenode.handler.count is 10 by default and will be increased to 50; the size of HDFS file block dfs.blocksize is 128MB by default and will be decreased to 64MB as the system contains all small files.

The function can only be performed once during startup of cluster.

How to use

Logon to Alibaba Cloud E-MapReduce Console Cluster List.

Select the region above and the created cluster appears in the corresponding region.

Click **Create cluster** to enter the cluster creation page.

All contained software and corresponding versions can be seen in the software configuration of cluster creation. Change the configuration of cluster by selecting corresponding json format configuration file in the (optional) software configuration box and override or adding the defaulted cluster parameters. The sample of a .json file is as below

```
{
"configurations": [
{
"classification": "core-site",
"properties": {
"fs.trash.interval": "61"
}
},
{
"classification": "hadoop-log4j",
"properties": {
"hadoop.log.file": "hadoop1.log",
"hadoop.root.logger": "INFO",
"a.b.c": "ABC"
}
```

```
},
{
"classification": "hdfs-site",
"properties": {
"dfs.namenode.handler.count": "12"
}
},
{
"classification": "mapred-site",
"properties": {
"mapreduce.task.io.sort.mb": "201"
}
},
{
"classification": "yarn-site",
"properties": {
"hadoop.security.groups.cache.secs": "251",
"yarn.nodemanager.remote-app-log-dir": "/tmp/logs1"
}
},
{
"classification": "httpsfs-site",
"properties": {
"a.b.c.d": "200"
}
},
{
"classification": "capacity-scheduler",
"properties": {
"yarn.scheduler.capacity.maximum-am-resource-percent": "0.2"
}
},
{
"classification": "hadoop-env",
"properties": {
"BC":"CD"
},
"configurations":[
{
"classification":"export",
"properties": {
"AB":"${BC}",
"HADOOP_CLIENT_OPTS":"\"-Xmx512m -Xms512m $HADOOP_CLIENT_OPTS\""
}
}
]
},
{
"classification": "httpfs-env",
"properties": {
},
"configurations":[
{
"classification":"export",
"properties": {
"HTTPFS_SSL_KEYSTORE_PASS":"passwd"
```

```
}
}
]
},
{
"classification": "mapred-env",
"properties": {
},
"configurations":[
{
"classification":"export",
"properties": {
"HADOOP_JOB_HISTORYSERVER_HEAPSIZE":"1001"
}
}
]
},
{
"classification": "yarn-env",
"properties": {
},
"configurations":[
{
"classification":"export",
"properties": {
"HADOOP_YARN_USER":"${HADOOP_YARN_USER:-yarn1}"
}
}
]
},
{
"classification": "pig",
"properties": {
"pig.tez.auto.parallelism": "false"
}
},
{
"classification": "pig-log4j",
"properties": {
"log4j.logger.org.apache.pig": "error, A"
}
},
{
"classification": "hive-env",
"properties": {
"BC":"CD"
},
"configurations":[
{
"classification":"export",
"properties": {
"AB":"${BC}",
"HADOOP_CLIENT_OPTS1":"\"-Xmx512m -Xms512m $HADOOP_CLIENT_OPTS1\""
}
}
]
```

```
},
{
"classification": "hive-site",
"properties": {
"hive.tez.java.opts": "-Xmx3900m"
}
},
{
"classification": "hive-exec-log4j",
"properties": {
"log4j.logger.org.apache.zookeeper.ClientCnxnSocketNIO": "INFO,FA"
}
},
{
"classification": "hive-log4j",
"properties": {
"log4j.logger.org.apache.zookeeper.server.NIOServerCnxn": "INFO,DRFA"
}
}
]
}
```

The classification parameter designates the configuration file to change; the properties parameter stores the key-value pair that requires changes; when the default configuration file has a corresponding key, override the value, otherwise, add the corresponding key-value pair.

The correspondence between configuration file and classification is shown in the table below:

Hadoop

Filename	classification
core-site.xml	core-site
log4j.properties	hadoop-log4j
hdfs-site.xml	hdfs-site
mapred-site.xml	mapred-site
yarn-site.xml	yarn-site
httpsfs-site.xml	httpsfs-site
capacity-scheduler.xml	capacity-scheduler
hadoop-env.sh	hadoop-env
httpfs-env.sh	httpfs-env
mapred-env.sh	mapred-env
yarn-env.sh	yarn-env

Pig

Filename	classification

pig.properties	pig
log4j.properties	pig-log4j

Hive

Filename	classification
hive-env.sh	hive-env
hive-site.xml	hive-site
hive-exec-log4j.properties	hive-exec-log4j
hive-log4j.properties	hive-log4j

The core-site and other flat xml files only have a layer. All configurations are put in properties. The hadoop-en v and other sh files may have two layers of structures and can be set in the embedded configurations mode. Refer to hadoop-env in the example where -Xmx512m -Xms512m setting is added for HADOOP_CLIENT_OPTS property of export.

After setting, confirm and click **Next step**.

Bootstrap Action

Function of bootstrap operation

The function of bootstrap operation is to execute your customized script before the cluster starts up Hadoop to install your required third-party software or change the cluster operating environment.

With bootstrap operation, you can install many things that are not supported by cluster currently to your cluster. For example:

Install provided software with yum.

Directly download some open software from public network.

Read your own data from OSS.

Install and operate a service such as Flink or Impala, but the script to be compiled will be more complex.

We strongly recommend you to test the bootstrap operation with the pay-as-you-go cluster and

create the subscription cluster after the test is passed.

How to use

Logon to Alibaba Cloud E-MapReduce Console Cluster List.

Select the region above and the created cluster appears in the corresponding region.

Click **Create cluster** to enter the cluster creation page.

Click **Add bootstrap operation** on the basic information page of cluster creation to enter the operation page.

Enter the configuration item on the page of adding bootstrap operation. Click Complete after adding.

You may add 16 bootstrap operations at most which will execute during cluster initialization in the sequence you designate. Your designated script will be executed with root account by default. You may switch to Hadoop account with su hadoop in the script.

The bootstrap operation may fail. For ease of use, the failure of bootstrap operation does not affect the creation of cluster. After the cluster is created successfully, you may view any abnormality in the **bootstrap/software configuration** of cluster information column on the cluster details page. In case of any abnormality, you can logon to all nodes to view the operation log in the directory of /var/log/bootstrap-actions.

Bootstrap operation type

The bootstrap operation is categorized into customized bootstrap operation and operating-condition bootstrap operation. The main difference is that the operating-condition bootstrap operation can only operate your designated operation in the node that meets requirement.

Customized bootstrap operation

For the customized bootstrap operation, the position of bootstrap operation name and execution script in OSS needs to be designated and the optional parameters will be designated as required. During cluster initialization, all nodes will download your designated OSS scripts to execute them directly or after adding the optional parameters.

You can designate the files that need to be downloaded from OSS in the script. The following

example will download the file oss://yourbucket/myfile.tar.gz locally and extract it to the directory of /yourdir:

```
#!/bin/bash
osscmd --id=<yourid> --key=<yourkey> --host=oss-cn-hangzhou-internal.aliyuncs.com get
oss://<yourbucket>/<myfile>.tar.gz ./<myfile>.tar.gz
mkdir -p /<yourdir>
tar -zxvf <myfile>.tar.gz -C /<yourdir>
```

The osscmd has been preinstalled on the node and can be invoked directly to download the file.

Note: OSS address host contains intranet address, internet address and VPC network address. For the classic network, the intranet address will be designated. The address in Hangzhou is oss-cn-hangzhou-internal.aliyuncs.com. For VPC network, the domain name that VPC intranet can visit will be designated. The name in Hangzhou is vpc100-oss-cn-hangzhou.aliyuncs.com.

The bootstrap operation can install additional system software package through yum. The following example is about installation of ld-linux.so.2:

```
#!/bin/bash
yum install -y ld-linux.so.2
```

Operating-condition bootstrap operation

The execution script of operating-condition bootstrap operation is predefined. You have no need to make additional designation. What you need do is to designate the name and optional parameters. The operating-condition bootstrap operation must provide the optional parameters which shall include the spaced operation conditions and commands. The operation conditions support instance.isMaster=true/false and is designated to operate only on the master or non-master nodes. The following example shows that the optional parameters of operating-condition bootstrap operation are designated to only create the directory on the master node.

```
instance.isMaster=true mkdir -p /tmp/abc
```

If multiple operation commands are designated, you may divide several statements with the semicolon ";" . For example:instance.isMaster=true mkdir -p /tmp/abc;mkdir -p /tmp/def

VPC

VPC (Virtual Private Cloud) creates an isolated network environment for users. You can select your own IP address range, divide network and configure the routing list and gateway. The interflow of VPC intranet and between VPC and physical IDC machine rooms can be realized among regions or

users.

Create a VPC cluster

E-MapReduce can select the type of network during cluster creation, namely classic network/VPC. For VPC, the following operations are required:

- Subordinate VPC: select which VPC the current E-MapReduce cluster is located. If no creation is made, enter VPC Console to create a VPC.

  Subnet (vswitch): ECS instance in E-MapReduce cluster communicates through vswtich. If no creation is made, enter VPC Console to create a VPC. The vswtich has the property of availability zone. Therefore, the created vswtich must also belong to the availability zone once the zone is selected during cluster creation in E-MapReduce.

  Safety group creation: Once enabled, you should enter the name of created security group.

  Owner security group: the security group to which the cluster belongs to. The security group of classic network cannot be used in VPC. The security group of VPC can only be used in current VPC. Here we will only show the security group that is created in E-MapReduce product by the user. For some safety reasons, the security group created outside the E-MapReduce cannot be selected. To create a security group, select **New security group** and enter the name of security group.

Example

E-MapReduce cluster communication in different VPCs (Hive visits HBase)

Create cluster:

Create two clusters in E-MapReduce. Hive Cluster C1 is located in VPC1 while HBase Cluster C2 is located in VPC2. Both clusters are located in cn-hangzhou region.

Configure the high-speed channel:

See VPC private network communication in the same region for configuration.

Log in to HBase cluster through SSH and a table is created through HBase Shell.

```
hbase(main):001:0> create 'testfromHbase','cf'
```

Log in to Hive through SSH

Change hosts and add a line as below:

```
$zk_ip emr-cluster //$zk_ip为Hbase集群的zk节点IP
```

Visit HBase through Hive Shell.

```
hive> set hbase.zookeeper.quorum=172.16.126.111,172.16.126.112,172.16.126.113;
hive> CREATE EXTERNAL TABLE IF NOT EXISTS testfromHive (rowkey STRING, pageviews Int, bytes STRING) STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES ('hbase.columns.mapping' = ':key,cf:c1,cf:c2') TBLPROPERTIES ('hbase.table.name' = 'testfromHbase');
```

The abnormality of java.net.SocketTimeoutException is reported at this moment because the security group where ECS of HBase cluster is located limits the E-MapReduce visit at relevant ports (Port 22 is only opened by default in the security group created by E-MapReduce). Therefore, the security group rules need to be added in the security group of HBase cluster to open port for Hive cluster, as shown in the figure below:

Authorization policy	Protocol type	Port range	Authorization type	Authorization object
Allow	TCP	2181/2181	Address field access	192.168.1.0/16
Allow	TCP	22/22	Address field access	0.0.0.0/0
Allow	TCP	16000/16000	Address field access	192.168.1.0/16
Allow	TCP	16020/16020	Address field access	192.168.1.0/16

Open source assembly

Instructions for Hue

E-MapReduce currently supports Hue. Hue can be accessed and used in E-MapReduce just by selecting a Hue-supported mirror image to create the cluster and enabling public network IP address.

Preparations

Before creating a cluster, it is required to enable SSH tunnel. For detailed steps, refer to Connect to

Cluster using SSH.

Taking the Mac environment as an example. Port forwarding is realized by the Chrome browser (assuming the IP address of the public network for the master node of cluster is “xx.xx.xx.xx”):

Connect to the master node.

```
ssh root@xx.xx.xx.xx
```

Enter a password.

Check id_rsa.pub content of local machine (note that this shall be executed on local machine rather than remote master node).

```
cat ~/.ssh/id_rsa.pub
```

Write id_rsa.pub content of local machine in ~/.ssh/authorized_keys on local master node (to execute on far-end master node).

```
mkdir ~/.ssh/
vim ~/.ssh/authorized_keys
```

Paste and save the content observed in Step 2. Now ssh root@xx.xx.xx.xx ought to be used directly to log on to the master node without password.

Execute commands below on local machine for port forwarding.

```
ssh -i ~/.ssh/id_rsa -ND 8157 root@xx.xx.xx.xx
```

Enable Chrome (to execute in the new terminal on local machine).

```
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --proxy-server="socks5://localhost:8157" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" --user-data-dir=/tmp
```

Access to Hue

In the Chrome browser for port forwarding, access: xx.xx.xx.xx:8888.

Instructions for Oozie

Version information

Alibaba Cloud E-MapReduce 2.0.0 and higher versions support Oozie. If it is necessary to use Oozie in a cluster, ensure the cluster version is higher than 2.0.0.

Preparations

Before creating a cluster, it is required to open SSH tunnel. Refer to Connect to Cluster using SSH for detailed steps.

Taking MAC environment as an example, port forwarding is realized by Chrome browser (assuming IP address of public network for master node of cluster is “xx.xx.xx.xx”):

Log on to the master node.

```
ssh root@xx.xx.xx.xx
```

Enter a password.

Check id_rsa.pub content of local machine (note that this shall be executed on local machine rather than remote master node).

```
cat ~/.ssh/id_rsa.pub
```

Write id_rsa.pub content of local machine in ~/.ssh/authorized_keys on local master node (to execute on far-end master node).

```
mkdir ~/.ssh/
vim ~/.ssh/authorized_keys
```

Paste the content observed in Step 2 and now ssh root@xx.xx.xx.xx can be used directly to log on to the master node without passwords.

Execute commands below on local machine for port forwarding.

```
ssh -i ~/.ssh/id_rsa -ND 8157 root@xx.xx.xx.xx
```

7. Enable Chrome (to execute in the new terminal on local machine).

```
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --proxy-server="socks5://localhost:8157" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" --user-data-dir=/tmp
```

Access Oozie UI interface

Access in Chrome browser for port forwarding: “xx.xx.xx.xx:11000/oozie”, “localhost:11000/oozie” or Intranet “ip: 11000/oozie”.

Submit workflow job

Since clusters with and without enabled HA have different modes to access NameNode and ResourceManager, when submitting oozie workflow job, it is required to specify different NameNode and JobTracker (ResourceManager) in job.properties files. Specific steps are as follows:

Non-HA cluster

```
nameNode=hdfs://localhost:9000
jobTracker=localhost:8032
```

HA cluster

```
nameNode=hdfs://emr-cluster
jobTracker=rm1,rm2
```

For operation examples below, configurations are made for both non-HA and HA cluster, that is to say, the sample code can be used directly for operation without any modification. For specific format of workflow file, refer to official document for Oozie at https://oozie.apache.org/docs/4.2.0/.

Submit workflow job on a non-HA cluster

Log on to the main master node of cluster.

```
ssh root@publicIp_of_master
```

Download sample code.

```
[root@emr-header-1 ~]# su oozie
[oozie@emr-header-1 root]$ cd /tmp
[oozie@emr-header-1 tmp]$ wget http://emr-sample-projects.oss-cn-hangzhou.aliyuncs.com/oozie-examples/oozie-examples.zip
[oozie@emr-header-1 tmp]$ unzip oozie-examples.zip
```

Synchronize Oozie workflow code to hdfs.

```
[oozie@emr-header-1 tmp]$ hadoop fs -copyFromLocal examples/ /user/oozie/examples
```

Submit Oozie workflow sample job.

```
[oozie@emr-header-1 tmp]$ $OOZIE_HOME/bin/oozie job -config examples/apps/map-reduce/job.properties -run
```

After successful execution, a jobId will be returned and is similar to:

```
job: 0000000-160627195651086-oozie-oozi-W
```

Visit Oozie UI page to see Oozie workflow job just submitted.

Submit workflow job on a HA cluster

Log on to the main master node of HA cluster.

```
ssh root@main_master_ip
```

Current main master node can be determined by checking whether the Oozie UI can be accessed or not, and the service of Oozie server is to be enabled on the main master node “xx.xx.xx.xx:11000/oozie” by default.

Download sample codes of HA cluster.

```
[root@emr-header-1 ~]# su oozie
[oozie@emr-header-1 root]$ cd /tmp
[oozie@emr-header-1 tmp]$ wget http://emr-sample-projects.oss-cn-hangzhou.aliyuncs.com/oozie-examples/oozie-examples-ha.zip
[oozie@emr-header-1 tmp]$ unzip oozie-examples-ha.zip
```

Synchronize Oozie workflow code to hdfs.

```
[oozie@emr-header-1 tmp]$ hadoop fs -copyFromLocal examples/ /user/oozie/examples
```

Submit Oozie workflow sample job.

```
[oozie@emr-header-1 tmp]$ $OOZIE_HOME/bin/oozie job -config examples/apps/map-reduce/job.properties -run
```

After successful execution, a jobId will be returned and is similar to:

```
job: 0000000-160627195651086-oozie-oozi-W
```

Visit Oozie UI page to see Oozie workflow job just submitted.

Instructions for Presto

E-MapReduce versions higher than 2.0 support presto, and presto can be used in E-MapReduce just by checking presto software when selecting the mirror image.

After cluster creation, log on to the master. Presto software is installed in the directory of /usr/lib/presto-current, and PrestoServer process can be seen by command jps.

Presto service process can be divided into coordinator and worker. Coordinator is started on the master (HA cluster is the master node of hostname started with emr-header-1), and worker process is started on the core node. The configuration of service process is under the directory of /usr/lib/presto-current/etc, in which, coordinator uses coordinator-config.properties while worker uses worker-config.preperties, and other configuration files are used for public. The web port is set as 9090.

Presto service is set with supports from Hive by default. Connect the metastore of hive on the cluster to read the table information of Hive and perform querying. The cluster is pre-installed with presto cli and can directly execute “show tables” to check the Hive list. Note that several seconds of delay will occur when synchronizing the Hive table.

```
presto -server localhost:9090 -catalog hive -schema default -user hadoop -execute 'show tables'
```

Instructions for Zeppelin

E-MapReduce currently supports Appache Zeppelin. Zeppelin can be accessed and used in E-MapReduce just by selecting a Zeppelin-supported mirror image to create a cluster and enabling

public network IP address.

Preparations

Before creating a cluster, it is required to enable SSH tunnel. For detailed steps, refer to: Connect to Cluster using SSH.

Take the Mac environment as an example. Using the Chrome browser with port dynamic forwarding (assuming the IP address of the public network for the master node of cluster is “xx.xx.xx.xx”):

Connect to the master node.

```
ssh root@xx.xx.xx.xx
```

Enter a password.

Check id_rsa.pub content of the local machine. Note that this shall be executed on the local machine rather than the remote master node.

```
cat ~/.ssh/id_rsa.pub
```

Write id_rsa.pub content of the local machine in ~/.ssh/authorized_keys on local master node (to execute on far-end master node).

```
mkdir ~/.ssh/
vim ~/.ssh/authorized_keys
```

Paste the content observed in Step 2.

ssh root@xx.xx.xx.xx

Now you can connect to the master node without password.

Execute the commands below on the local machine for port forwarding.

```
ssh -i ~/.ssh/id_rsa -ND 8157 root@xx.xx.xx.xx
```

Enable Chrome (to execute in a new terminal on the local machine).

```
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --proxy-
```

```
server="socks5://localhost:8157" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" --user-data-dir=/tmp
```

Access to Zeppelin

In the Chrome browser for port forwarding, access “xx.xx.xx.xx:8080”.

Instructions for ZooKeeper

The ZooKeeper service is currently enabled in E-MapReduce cluster by default.

Considerations

ZooKeeper will have only 3 nodes no matter how many machines are currently in the cluster. More nodes are not supported currently.

Create a cluster

Zookeeper will be ticked by default in the software configuration page where the E-MapReduce creates the cluster.

Node information

After the cluster is successfully created and the status is idle, view the cluster details page and you will find the node information of ZooKeeper. Corresponding Intranet IP address (2181 for port by default) of ZooKeeper node is indicated on the process column for the access to ZooKeeper services.

Version list is shown as below:

Version	Containing library	Installation position
2.6		
2.7.11	numpy	/usr/local/Python-2.7.11/
3.4.4	numpy	/usr/local/Python-3.4.4/

Please enter the full path of the Python command for the corresponding version of the script for use.

Connect to the Cluster Using SSH

Use of connect to the cluster

If you think the job and execution plan on the page cannot meet your more complex application requirements, you may log in to the master of E-MapReduce cluster to find the cluster details page where there is the public network IP address of the cluster master. You can directly log in to the machine through SSH to view various settings and states.

Relevant environment variables have been set on the machine, including the following common ones:

JAVA_HOME

HADOOP_HOME

HADOOP_CONF_DIR

HADOOP_LOG_DIR

YARN_LOG_DIR

HIVE_HOME

HIVE_CONF_DIR

PIG_HOME

PIG_CONF_DIR

You can directly quote these variables in the script but never change them to avoid unexpected errors of E-MapReduce.

Connect to Master

SSH logs in to the master with the following commands. Please obtain the public network IP of cluster master in the hardware information column of Cluster Details Page.

```
ssh root@ip.of.master
```

Enter the password set during creation.

Connect to Cluster Using SSH without password

You need to connect to the cluster generally for some management and operation. To connnect to the cluster master, you can break through the SSH password-less login from the master machine (the cluster master opens up the public network IP by default). Operation steps are as follows:

connect to the master with the root and password mode as mentioned above.

change to **Hadoop** or **hdfs** user. su hadoop

Connect to the Master Node Using SSH on Linux, Unix, and Mac OS X

Copy the private key to local machine.

```
rz ~/.ssh/id_rsa
```

Return to your local machine and try to connect to the master again.

```
ssh -i private_key_path/id_rsa hadoop@server_ip_address
```

If there is only one private key, you may put it in your ~/.ssh/ and use it by default without designation of -i.

Connect to the Master Node Using SSH on Windows

You can connect to the master through SSH without input password with multiple methods under Windows.

Method I: PuTTY

Click Download PuTTY.

Download PuTTYgen from the same location.

Open PuTTYgen and load your private key.

Note: Please keep the private key safe. In case of disclosure, please generate a new private key immediately for replacement.

Use default configuration and save the private key. Obtain a secret key file of PuTTY with suffix of ppk.

Operate PuTTY and select Session on the configuration page.

Enter the public network IP address of target machine you will connect and add the user name like hadoop@MasterNodeIP.

Select Connection on the configuration page. Unfold > select SSH and unfold > select Auth.

Select the generated ppk file.

Click Open to log in to the master node automatically.

Method II: Cygwin | MinGW

It is a very convenient tool to simulate Linux env in Windows.

For this method, refer to above SSH method of Linux.

MinGW method is recommended for use which is the most compact. Download git client if the official website cannot be opened. The default Git Bash can be used.

View webui of Hadoop, Spark, Ganglia and other systems

Note: Please confirm you have finished above SSH password-less login process before this step.

For safety, the webui monitoring system ports of Hadoop、Spark, Ganglia and other systems in the E-MapReduce cluster are not opened to the outside world. If you want to visit these webUIs, a SSH tunnel needs to be built to forward through port. Two following methods are available:

Note: The following operations are completed in your local machine, instead of the machine in the cluster.

Method I: Port dynamic forwarding

Create a SSH tunnel which can connect certain dynamic port connection between local machine and master machine in E-MapReduce cluster.

```
ssh -i /path/id_xxx  -ND 8157 hadoop@masterNodeIP
```

8157 is any port not used in the local machine and can be customized by you.

After dynamic forwarding, you can view in the following .

Recommended methods

Recommend to use Chrome browser. Visit Web UI in the following methods:

The location of Chrome varies with operating systems. Refer to the table below:

Operating System	Chrome Location
Mac OS X	/Applications/Google Chrome.app/Contents/MacOS/Google Chrome
Linux	/usr/bin/google-chrome
Windows	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe

```
chrome --proxy-server="socks5://localhost:8157" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" --user-data-dir=/tmppath/
```

replace chrome with real location on the above table.

For Windows system, the tmppath can be written into similar d:/tmppath. For Linux or OSX, /tmp/ can be written directly.

Method II: Local port forwarding

Note: Its disadvantage is that only the interface in the outermost layer can be seen. The view of detailed job information will result in error.

```
ssh -i /path/id_rsa -N -L 8157:masterNodeIP:8088 hadoop@masterNodeIP
```

Parameter description:

path: private key storage path.

masterNodeIP: IP address of the master node to be connected.

8088: access port number of ResourceManager on the master node.

MetaService is provided under the E-MapReduce environment. MetaService allows you to access Alibaba Cloud resources in the E-MapReduce cluster without using AK.

Default Roles

By default, you need to authorize an application role (AliyunEmrEcsDefaultRole) to E-MapReduce when creating a cluster. After authorization, you can perform operations on E-MapReduce to access Alibaba Cloud resources without needing to explicitly input AK. By default, the following permission policies are granted to AliyunEmrEcsDefaultRole:

```
{
"Version": "1",
"Statement": [
{
"Action": [
"oss:GetObject",
"oss:ListObjects",
"oss:PutObject",
"oss:DeleteObject",
"oss:ListBuckets",
"oss:AbortMultipartUpload"
],
"Resource": "*",
"Effect": "Allow"
}
]
}
```

By default, your operations based on MetaService will be able to **Access OSS Data Only**. If you want to access other Alibaba Cloud resources like LogService by using MetaService, you need to grant permissions to AliyunEmrEcsDefaultRole. You can perform the preceding operations by using the RAM Console.

NOTE: Currently, MetaService only supports AK-free operations on the OSS, LogService and MNS data. You must **edit and delete the default role with caution**. Otherwise, your cluster creation or operations will fail.

Custom Application Role

In most cases, you only need to use a default role or modify the default role. E-MapReduce also allows you to create your own application role. When creating a cluster, you can use a default role or create your own application role. For details about how to create and authorize a role to E-MapReduce, refer to the RAM Documentations.

Accessing MetaService

MetaService is an HTTP service which can be accessed directly to obtain meta information. For example, you can access “curl http://localhost:10011/cluster-region ” to view the region where the current cluster is located.

Currently, MetaService supports the following types of information:

- Region： “/cluster-region”
- Role name: “/cluster-role-name”
- AccessKeyId：” /role-access-key-id”
- AccessKeySecret：” /role-access-key-secret”
- SecurityToken：” /role-security-token”
- Network type: “/cluster-network-type”

Using MetaService

You can use MetaSerivce to access Alibaba Cloud resources without needing to use AK, which can:

- Reduce the risk of AK leak. The RAM-based usage can minimize security risk. The permissions are minimized through granting only the required permissions to the role.
- Improve user experience. Especially when you interactively access the OSS resources by using MetaService, you do not need to write a long string of OSS path.

Several usage methods are introduced below:

```
I. Using the Hadoop command line to display OSS data

Previously, we used: hadoop fs -ls oss://ZaH******As1s:Ba23N**********sdaBj2@bucket.oss-cn-hangzhou-internal.aliyuncs.com/a/b/c
Now, we use: hadoop fs -ls oss://bucket/a/b/c

II. Using Hive to create a table

Previously, we used:
CREATE EXTERNAL TABLE test_table(id INT, name string)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '/t'
LOCATION 'oss://ZaH******As1s:Ba23N**********sdaBj2@bucket.oss-cn-hangzhou-internal.aliyuncs.com/a/b/c';
Now, we use:
CREATE EXTERNAL TABLE test_table(id INT, name string)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '/t'
LOCATION 'oss://bucket/a/b/c';

III. Spark

Previously, we used: val data = sc.textFile("oss://ZaH******As1s:Ba23N**********sdaBj2@bucket.oss-cn-hangzhou-internal.aliyuncs.com/a/b/c")
```

```
Now, we use: val data = sc.textFile("oss://bucket/a/b/c")
```

Notebook

Notebook allows you to compile and run Spark, Spark SQL and Hive SQL tasks directly on the E-MapReduce console. You can view the running results directly on the Notebook. The Notebook is ideal for processing debugging tasks that require shorter runtime and whose data results need to be viewed directly. For tasks with longer runtime and requiring regular execution, the job function and execution plan function must be used.

This section describes how to create and run a demo task. For other examples and operation instructions, refer to the subsequent chapters.

Creating a Demo Task

Log in to the Alibaba Cloud E-MapReduce Console Interactive Work.

Click **New notebook demo**

A confirmation box is displayed, indicating the cluster environment required. Click “OK” to create demo tasks. Three examples of interactive tasks will be created.

Running a Spark Demo Task

Click **EMR-Spark-Demo** to display the example of a Spark notebook. Before running the notebook, you need to associate the task to a created cluster first. Click to select a created cluster in the list of available clusters. Note that the associated cluster must be EMR-2.3 or later, and has no less than three nodes, each with at least 4 cores and 8GB of memory.

After a cluster is associated, click **Run**. When the associated cluster executes the Spark or Spark SQL notebook for the first time, it takes about one minute to build the Spark context and running environment. It does not need to be built in subsequent executions. The running result is displayed below the **Run** button.

Running a SparkSQL Demo Task

Click **EMR-Spark-Demo** to display the example of a SparkSQL notebook. Before running the notebook, you also need to associate the notebook to a created cluster first. Click the top right corner and select a created cluster in the list of available clusters.

The SparkSQL demo contains several demo sections, which can be run individually or wholly by clicking **Run All**. After running, you can see all returned data results of each section. Note that if the section for creating a table is run multiple times, an error will be reported indicating that the table has already existed.

Running a Hive Demo Task

Click **EMR-Hive-Demo** to display the example of a Hive notebook. Before running the notebook, you also need to associate the notebook to a created cluster first. Click the top right corner and select a created cluster in the list of available clusters.

The Hive demo task contains several demo sections, which can be run individually or wholly by clicking **Run All**. After running, you can see all returned data results of each section. NOTE 1: When the associated cluster executes the Hive notebook for the first time, it takes about tens of seconds to build the Hive client running environment. It will no longer need to be built in subsequent execution. NOTE 2: If the section for creating a table is run multiple times, an error will be reported indicating that the table has already existed.

Cancelling Association with Clusters

After a notebook is run in a cluster, the cluster creates a process for catching of some context running environments in order to ensure quick response upon re-execution. If you do not need to execute other notebooks and you want to release the cluster resources occupied by caching, you can disassociate all interactive tasks that have been run from the associated clusters. In this way, you can release the memory resources occupied on the original associated clusters.

Creating an Notebook

NOTE: The cluster on which an interactive task is run must be EMR-2.3 or later, and has no less than three nodes, each with at least 4 cores and 8GB of memory.

Log in to the Alibaba Cloud E-MapReduce Console Interactive Work.

Click **New notebook** or **File** -> **New notebook**.

Fill in the name and select the default type. The associated cluster is optional. Click “OK” to create a notebook.

Currently, three types are supported. **Spark** can be used to write scala spark codes. **Spark SQL** can be used to write SQL statements supported by Spark. **Hive** can ne used to write SQL statements supported by Hive.

An associated cluster must be a created cluster of EMR-2.3 or later, and has no less than three nodes, each with at least 4 cores and 8GB of memory. You can also associate the cluster only immediately before running the task.

Currently, up to 20 interactive tasks can be created in one account.

Filling In and Saving a Section

A paragraph is the smallest unit for running a notebook. For a notebook, you can fill in multiple paragraphs. Each paragraph can start with %spark, %sql, or %hive, indicating that this paragraph is a Scala spark code paragraph, spark SQL paragraph or Hive SQL paragraph. The prefix of type is segregated by a blank space or by line feed and actual content. If the prefix of type is not specified, the default type of the interactive task will be used as the run type of this paragraph.

The following is an example showing how to create a temporary Spark table:

Paste the following code in the section and a red * symbol is displayed, indicating that this notebook has been changed. You can click the **save paragraph** button or **run** button to save the modifications to the paragraph. Click “+” below the paragraph to create a new paragraph.Currently, up to 30 paragraphs can be created in one notebook.

```
%spark
import org.apache.commons.io.IOUtils
import java.net.URL
import java.nio.charset.Charset

// load bank data
val bankText = sc.parallelize(
IOUtils.toString(
new URL("http://emr-sample-projects.oss-cn-hangzhou.aliyuncs.com/bank.csv"),
Charset.forName("utf8")).split("\n"))
case class Bank(age: Integer, job: String, marital: String, education: String, balance: Integer)
val bank = bankText.map(s => s.split(";")).filter(s => s(0) != "\"age\"").map(
s => Bank(s(0).toInt,
s(1).replaceAll("\"", ""),
```

```
s(2).replaceAll("\"", ""),
s(3).replaceAll("\"", ""),
s(5).replaceAll("\"", "").toInt
)
).toDF()
bank.registerTempTable("bank")
```

Running a Paragraph

Before running a notebook, you must associate it to a created cluster first. If a created notebook is not associated with a cluster, **Not associated** is displayed on the top right corner of the page. You can click to select a cluster in the list of available clusters. Note that the associated cluster must be EMR-2.3 or later, and has no less than three nodes, each with at least 4 cores and 8GB of memory.

Click the **Run** button to automatically save the current paragraph and run the content. If this paragraph is the last one, a new paragraph is automatically created.

After running, the current running status is displayed. PENDING means the paragraph has not been run yet and RUNNING means the paragraph is running. FINISHED means the running process has been finished. ERROR means an error occurs. The running result is displayed below the run button of the paragraph. During running, you can click **Cancel** below the **Run** button to cancel running. ABORT is displayed after running is canceled.

The paragraph can be run multiple times and only the result of the last running is retained. You cannot modify the entered content of the paragraph during running. The content can be modified only after running of the paragraph is finished.

Running All

For a notebook, you can click **Run All** on the menu bar to run all paragraphs. The paragraphs are submitted sequentially for running. Different types have independent execution queues. If a notebook contains multiple paragraph types, the order for executing the paragraphs on the cluster is based on type after these paragraphs are submitted sequentially. Spark and Spark SQL support one-by-one execution. Hive supports concurrent execution and the maximum number of concurrently executed interactive paragraphs on the same cluster is 10. Note that all concurrently executed paragraphs are restricted by cluster resources. If the cluster size is small and many paragraphs need to be executed concurrently, the paragraphs still need to queue on the Yarn.

Cancelling Association with Clusters

After an notebook is run in a cluster, the cluster creates a process for catching of some context running environments in order to ensure quick response upon re-execution. If you do not need to execute other notebooks and you want to release the cluster resources occupied by caching, you can disassociate all notebooks that have been run from the associated clusters. In this way, you can release the memory resources occupied on the original associated clusters.

Other Operations

Paragraph Operations

Hiding and Displaying Results

You can hide the paragraph results and display the entered content of the paragraph only.

Deleting a Paragraph

Delete the current paragraph. The paragraph that is running can also be deleted.

File Menu

New notebook

Create a notebook, and switch to the created notebook interface.

Creating a Paragraph

Add a new paragraph to the end of the notebook. A notebook can have only up to 30 paragraphs.

Saving All Paragraphs

Save all modified paragraphs.

Deleting an Notebook

Delete the current notebook. If the cluster has been associated, it will also be disassociated.

View

Displaying Codes Only or Displaying Codes and Results

Only the entered codes for all paragraphs are displayed, or both the codes and results are displayed.

Examples

Section 1: Create a temporary table

```
%spark
import org.apache.commons.io.IOUtils
import java.net.URL
import java.nio.charset.Charset

// Zeppelin creates and injects sc (SparkContext) and sqlContext (HiveContext or SqlContext)
// So you don't need create them manually
// load bank data
val bankText = sc.parallelize(
IOUtils.toString(
new URL("http://emr-sample-projects.oss-cn-hangzhou.aliyuncs.com/bank.csv"),
Charset.forName("utf8")).split("\n"))
case class Bank(age: Integer, job: String, marital: String, education: String, balance: Integer)
val bank = bankText.map(s => s.split(";")).filter(s => s(0) != "\"age\"").map(
s => Bank(s(0).toInt,
s(1).replaceAll("\"", ""),
s(2).replaceAll("\"", ""),
s(3).replaceAll("\"", ""),
s(5).replaceAll("\"", "").toInt
)
).toDF()
bank.registerTempTable("bank")
```

Section 2: Query the table structure

```
%sql
desc bank
```

Section 3: Query the number of employees of each age group below 30

```
%sql select age, count(1) value from bank where age < 30 group by age order by age
```

Section 4: Query the information of employees at the age less than or equal to 20

```
%sql select * from bank where age <= 20
```

Data preparation

In this example, you need to download data from OSS and upload them to your own OSS bucket.

The data includes

- User Table Sample Data

- Video Table Sample Data
- Playvideo Table Sample Data

Upload the sample data of User Table, Video Table, and Playvideo Table to the specified UserInfo, Videoinfo, and Playvideo respectively on your OSS bucket. For example, upload the data to Demo or UserInfo directory under Bucket Example.

In the table created as below, replace the SQL [bucketname] with your bucket name like “example”, replace [region] with your OSS region name like “Hangzhou”, and replace [bucketpath] with your specified path prefix of the OSS, for example, “Demo”.

Section 1: Create a user table

```
%hive
CREATE EXTERNAL TABLE user_info(id int,sex int,age int, marital_status int) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LOCATION 'oss://[bucketname].oss-cn-[region]-internal.aliyuncs.com/[bucketpath]/userinfo'
```

Section 2: Create a video table

```
%hive
CREATE EXTERNAL TABLE video_info(id int,title string,type string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LOCATION 'oss://[bucketname].oss-cn-[region]-internal.aliyuncs.com/[bucketpath]/videoinfo'
```

Section 3: Create a video play table

```
%hive
CREATE EXTERNAL TABLE play_video(user_id int,video_id int, play_time bigint) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LOCATION 'oss://[bucketname].oss-cn-[region]-internal.aliyuncs.com/[bucketpath]/playvideo'
```

Section 4: User table count

```
%sql select count(*) from user_info
```

Section 5: Video table count

```
%sql select count(*) from video_info
```

Section 6: Video play table count

```
%sql select count(*) from play_video
```

Section 7: Video play count of each video type

```
%sql select video.type, count(video.type) as count from play_video play join video_info video on (play.video_id = video.id) group by video.type order by count desc
```

Section 8: Video information of video play top 10

```
%sql select video.id, video.title, video.type, video_count.count from (select video_id, count(video_id) as count from play_video group by video_id order by count desc limit 10) video_count join video_info video on (video_count.video_id = video.id) order by count desc
```

Section 9: Age groups of audience watching the video with largest video play count

```
%sql select age , count(*) as count  from (select distinct(user_id) from play_video  where video_id =49 ) play join user_info userinfo on (play.user_id = userinfo.id) group by userinfo.age
```

Section 10: Gender, age group, and marital status of audience watching the video with largest video play count

```
%sql select if(sex=0,'Female','Male') as title, count(*) as count, 'Gender' as type from (select distinct(user_id) from play_video  where video_id =49 ) play join user_info userinfo on (play.user_id = userinfo.id) group by userinfo.sex
union all
select case when userinfo.age<15 then 'Less than 15' when age<25 then '15-25' when age<35 then '25-35' else 'More than 35' end , count(*) as count, 'Age Group' as type from (select distinct(user_id) from play_video where video_id =49) play join user_info userinfo on (play.user_id = userinfo.id) group by case when userinfo.age<15 then 'Less than 15' when age<25 then '15-25' when age<35 then '25-35' else 'More than 35' end
union all
select if(marital_status=0,'Unmarried','Married') as title, count(*) as count, 'Marital Status' as type from (select distinct(user_id) from play_video where video_id =49 ) play join user_info userinfo on (play.user_id = userinfo.id) group by marital_status
```