# Analytic DB

## **Quick Start**

MORE THAN JUST CLOUD | C-) Alibaba Cloud

## **Quick Start**

### 1.1 Opening Analytic DB Service

On the public cloud, users that meet the opening conditions can apply to open the service at http://buy.aliyun.com/ads.

#### **1.2 Creating Databases**

In Analytic DB, you must create a database through the iDB Cloud Console page.

In the current Analytic DB version, during database creation, you must enter the database name. Note that this database name must be globally unique within the Analytic DB region. Then, select the Analytic DB region in which it is located, e.g. Hangzhou, Beijing.

Next, you must select this database' s type in Analytic DB. Analytic DB will optimize the database based on its type. There are three types:

App Type: This type is suitable for development of applications where the queries are relatively simple, there are no multi-table join operations and not much result data will be returned from calculations. Generally, it has a high number of QPS. The response time (RT) for such applications is normally less than 500 milliseconds.

BI Type: This type is suitable for use as a real-time data warehouse in BI type applications where a fact table is used as the core, multiple dimension tables and fact tables are joined, and there are many grouping and sorting operations. Generally, it has a low number of QPS (20 or less) and the response time (RT) for such applications is normally under 4000 milliseconds.

Adhoc Type: This type is suitable for use in complex analytic applications where typically multiple fact tables are joined, grouped, sorted, etc., or a large number of query results (500+ entries) are returned. Generally, it has a number of QPS equal to or slightly higher than the BI type. However, the response time (RT) for such applications is normally under 8000 milliseconds.

After entering all the options, click Create Database. If an error is returned, fix the problem based on

the error prompt (normally it is because the database name is already in use or is non-compliant). Otherwise, creation is successful. Within ten minutes, the iDB Cloud interface will display the link for the new database.

#### 1.3 Using DMS to Create and Manage Tables

Earlier, we had created an Analytic DB database. Analytic DB uses relational models to store data and uses 2D tables for data organization and storage. Like MySQL, before the data is imported into Analytic DB, you must establish a corresponding data table. Furthermore, in order to manage the relevant data tables, Analytic DB has introduced the table group concept.

Table groups are entities one level below databases and one level above tables. In Analytic DB, a table must belong to a table group. We will give a more detailed description of table groups in 3.2. Here, we will first create a table group.

In DMS for Analytic DB, right-click on the table group object on the left side and select New Table Group to bring up the Create New Table Group dialog box. The content of this dialog box is shown in the image below. We will enter the table group name test\_group and provisionally use the default values for all the other parameters.

新建表组		$\otimes$
表组名:	test_group	]
最小副本数:	2 -	]
超时时间(毫秒):	30000 ‡	]
	提办 取消	

After clicking OK to create a table group, we right-click on the table group and select New Table. Here, based on the test data, we create a data table with five columns and a primary partition. This is shown below:

DMS For ADS 1.0	新建 ~   SQLt	窗口	性能报告 导入导	出~ 新建授	汉		gq5Fl	DS2IgSWqXzTu →	- 使用手册
对象列表 ③	首页 新聞	衷 ×	新建授权 ×						
base	新增列 副	徐列	上移列 下移列			列属性			
base_dimension_grou	列名	Ŧ	数据类型 🔻	是否主键 * 是	否可为NULL 🔻 默认值	索引方3	t:	HASH索引	*
i i i i i i i i i i i i i i i i i i i	1 user_id 2 amt		bigint			列注释:			
🔤 table2 🔄 test1	3 num		int						
並 🐥 用户	4 cat_id 5 thedate		int	2					
	* <b>*</b> **					0 ()7			0
	-20(76)王								
	表组: 是否维度表:	hehe	Ŧ	表名: 更新方式:	my_first_table 实时更新	<ul> <li>一级分区</li> <li>分区方式</li> </ul>	[列: user_id : HASH	▽ 哈希分区数:	- 100 ‡
	聚集列:			注释:	我的第一个ADS表	二級分区	:		
	查询超时时间(毫秒):	30000	¢		_				
					保存				

In Analytic DB, data are stored and calculated in a distributed manner based on the partitioning column. As an example, here we distribute the raw data relatively uniformly by user\_id. Therefore, we specify hash partitions by user\_id and adjust the number of hash partitions to 40 (generally, the data of each partition should not exceed 8 million lines, though this is not an absolute limit. The number of partitions cannot exceed 256). The table and column names are entered based on the actual situation. Currently, the names must be consistent with those in the source data table.

In addition, if the data for this table is batch imported from another system (e.g. from ODPS), select Batch Update for the update method. Please refer to Importing Data later in Section 1.4. If the data for this table is directly inserted, select Real-time Update for the update method. Please refer to Inserting Data later in Section 1.4.

Analytic DB has a powerful automatic indexing function. When creating tables, users generally do not have to worry about column indexing. Analytic DB will perform automatic indexing based on the distribution of the actual data. Therefore, we will not adjust the column indexing settings here. Any table name can be entered and, for the table group name, we will select the test\_group table group we just created. Then, click Save to bring up the actual table creation DDL for verification.

We will provide detailed information about tables and columns a bit later in Sections 3.3 and 3.4. After table creation, right-click on the existing table to edit it.

### 1.4 Importing Data

If the selected data source is batch import when a table is created, the Analytic DB will provide multiple data import methods, such as the data pipeline command series (see 5.1) and Rest-like API methods. Here, as a test, we will import data through the iDB Cloud interface.

Prior to importing data, we must authorize the data source table, e.g. the data source table on ODPS. If on a public cloud, we must grant the describe and select permissions for garuda\_build@aliyun.com on ODPS (each private cloud authorized account name references the relevant configuration file of the private cloud and is not necessarily this account). In addition, please note that the Analytic DB currently only permits the operator to import data from ODPS projects of which he is the Project

Owner or ODPS tables of which he is the Table Creator.

Enter the DMS page and select the Import button on the menu bar to bring up the Import dialog box. Here, our data source table is on AliCloud ODPS. Therefore, the data import path follows the format

"odps://project\_name/table\_name/partition\_spec" . Concerning the partition information of the imported data, the iDB Cloud will automatically identify and enter this information in case that only Hash partitions exist. After entering this information, as shown in the image below, click "OK" .

导入数据					
数据源:	odps 📼	是否覆盖: 否 🔹			
数据源路径:	odps://ads_test_project/my_first_table/pt=2014				
目标表:	my_first_table				
一级分区列:	USER_ID				
填写版本号:	◎是	• 否			
	确定	关闭			

The following page will show the import statuses and the Analytic DB will schedule the import tasks. Based on how busy the system is and the amount and structure of the data to be imported, data import may take 20 minutes to several hours.

If the data source is set to real-time writing at the creation of the table, you may directly write SQL in the SQL window after the table is created:

insert into table my\_first\_table (user\_id,amt,num,cat\_id,thedate) values (12345, 80, 900, 1555, 20140101);

Right after a table has been created for real-time writing, it will need about ten minutes of preparation. The data written to the table at this time can only be queried at the end of the preparation. Otherwise, an error will be returned. After preparation is complete, the data changes due to the real-time insertion or deletion of data can generally be viewed after a maximum delay of one minute.

You must note that, when the Analytic DB performs real-time insertion or deletion, transactions are not supported and only the eventual consistency of the design is followed. Therefore, the Analytic DB cannot be used as an OLTP system.

If the table data is batch imported offline but the data source is RDS or another cloud system, we can synchronize the data through the AliCloud CDP product. After activating CDP on http://www.aliyun.com/product/cdp/ (you may have to apply for public beta), follow the sample below to configure the synchronization Job:

```
{
"type": "job",
"traceId": "rds to ads job test",
"version": "1.0",
"configuration": {
"setting": {
},
"reader": {
"plugin": "mysql",
"parameter": {
"instanceName": "Your RDS instance name",
"database": "RDS database name",
"table": "RDS table name",
"splitPk": "The name of any column",
"username": "RDS user name",
"password": "RDS password",
"column": ["*"],
}
},
"writer": {
"plugin": "ads",
"parameter": {
"url": "Connection information provided when the database is selected on the Analytic DB console",
"schema": "Analytic DB database name",
"table": "Analytic DB table name",
"username": "Your access key id",
"password": "Your access key secret",
"partition": "",
"lifeCycle": 2,
"overWrite": true
}
}
}
}
```

You must note that, before running this task, you must at least grant the Load Data permission on the table for cloud-data-pipeline@aliyun-inner.com in the Analytic DB.

#### 1.5 Connecting to and Using Analytic DB in Applications

After successfully importing data to the Analytic DB, we may wish that our application system would be able to connect to Analytic DB to query data. Any client that supports 5.1.x, 5.4.x, or 5.6.x protocol may connect to the Analytic DB. The domain name and port number used by the connection can be

viewed in the top-right corner of the iDB Cloud interface. The username and password used for the connection are the user' s AliCloud Access Key, which can be viewed and managed on https://i.aliyun.com/access\_key/. The Access Key ID is the user name and the Access Key Secret is the password (Analytic DB promises not to save user' s Access Key information).

#### Connecting to Analytic DB in PHP

In the PHP environment, assuming we have installed the module php-mysql 5.1.x (php\_MySQL.dll in Windows), we will create an ads\_conn.php. The content is as follows:

```
$ads_server_name="mydbname.ads-cn-hangzhou-1.aliyuncs.com "; //Database server name
$ads_username="my_access_key_id"; // Database connection username
$ads_password="my_access_key_secret"; // Database connection password
$ads_database="my_ads_db"; // Database name
$ads_port=3003; //Database port number
// Connected to database
$ads_conn=mysqli_connect($ads_server_name, $ads_username, $ads_password, $ads_database, $ads_port);
```

When querying, you can use:

```
$strsql="SELECT user_id FROM my_ads_db.my_first_table limit 20;"; $result=mysqli_query($ads_conn, $strsql);
```

while(\$row = mysqli\_fetch\_array(\$result)) {
 echo \$row["user\_id"]; //user\_id as the column name
}

The code given above can retrieve and print any ten recorded user\_ids. Note that, for data queries, Analytic DB does not support the SELECT \* method to query all columns.

#### Connecting to Analytic DB in JAVA

In JAVA, we will normally connect to the Analytic DB through a connection pool. Here, we use the domestic high-performance connection pool Druid to demonstrate how to connect to Analytic DB.

```
import com.alibaba.druid.pool.*;
DruidDataSource dataSource = new DruidDataSource();
dataSource.setDriverClassName("com.mysql.jdbc.Driver");
dataSource.setUsername("my_access_key_id");
dataSource.setPassword("my_access_key_secret");
dataSource.setUrl("jdbc:mysql://mydbname.ads-hz.aliyuncs.com:5544/my_ads_db");
// Connections configuration
dataSource.setInitialSize(5);
dataSource.setMinIdle(1);
dataSource.setMinIdle(1);
dataSource.setMaxActive(10);
// Enable monitoring statistics
dataSource.setFilters("stat");
// for mysql
dataSource.setPoolPreparedStatements(false);
// Use heartbeat statement to detect idle connections
```

dataSource.setValidationQuery('show status like "%Service\_Status%";'); dataSource.setTestWhileIdle(true);

In the above, please note that, if you must use heartbeat SQL for Analytic DB service status detection in any language, please use the "show status like '%Service\_Status%'" statement. If one line with two columns are returned, and the second column is 1, the Analytic DB service is working properly.