

云数据库 MongoDB

最佳实践

产品文档



腾讯云

【版权声明】

©2013-2018 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

最佳实践

代码示例

重连机制

导入导出

最佳实践

代码示例

最近更新时间：2017-12-21 17:53:39

以Python代码示例来演示对DynamoDB实例的基础访问和数据读写，实例创建完成之后

示例代码：

```
#!/usr/bin/python
```

```
from __future__ import print_function # Python 2/3 compatibility
import boto3
import json
import decimal
import time
from boto3.dynamodb.conditions import Key, Attr
```

```
#dynamodb = boto3.resource('dynamodb', region_name='us-west-2', endpoint_url="http://10.247.10.101:9000")
dynamodb = boto3.resource('dynamodb', region_name='us-west-2', endpoint_url="http://10.112.112.228:8000")
```

```
table = dynamodb.create_table(
    TableName='music',
    KeySchema=[
        {
            'AttributeName': 'Artist',
            'KeyType': 'HASH' #Partition key
        },
        {
            'AttributeName': 'SongTitle',
            'KeyType': 'RANGE' #Sort key
        },
    ],
    AttributeDefinitions=[
        {
            'AttributeName': 'Artist',
            'AttributeType': 'S'
        },
        {
            'AttributeName': 'SongTitle',
            'AttributeType': 'S'
        },
    ],
```

```
],

GlobalSecondaryIndexes=[
{
  "IndexName": "t1Index",
  "KeySchema": [
    {
      "AttributeName": 'Artist',
      "KeyType": "HASH"
    },
    {
      "AttributeName": 'SongTitle',
      "KeyType": "HASH"
    },
  ],
  "Projection": {
    "ProjectionType": "KEYS_ONLY"
  },
  "ProvisionedThroughput": {
    'ReadCapacityUnits': 3,
    'WriteCapacityUnits': 3
  }
}
],

ProvisionedThroughput={
  'ReadCapacityUnits': 3,
  'WriteCapacityUnits': 3
}
)

print("Table status:", table)
#print "Table status: ",table

time.sleep(30)
#write data
response = dynamodb.batch_write_item(
  RequestItems={
    'music': [
      {
        'PutRequest': {
          'Item': {
            "Artist": "The Acme Band 2",
            "SongTitle": "Look Out, World",
```

```
"AlbumTitle": "The Buck Starts Here",
"Price": decimal.Decimal('0.99'),
"Genre": "Rock",
}
},
{
  'PutRequest': {
    'Item': {
      "Artist": "none know",
      "SongTitle": "test World",
      "AlbumTitle": "The Buck Starts Here",
      "Price": decimal.Decimal('0.99'),
      "Genre": "Rock",
    }
  },
  {
    'PutRequest': {
      'Item': {
        "Artist": "The Acme Band 4",
        "SongTitle": "Look Out, World",
        "AlbumTitle": "The Buck Starts Here",
        "Price": decimal.Decimal('3.99'),
        "Genre": "Rock 4",
      }
    },
    {
      'PutRequest': {
        'Item': {
          "Artist": "The Acme Band 5",
          "SongTitle": "Look Out, World",
          "AlbumTitle": "The Buck Starts Here 5",
          "Price": decimal.Decimal('5.99'),
          "Genre": "Rock",
        }
      }
    },
  ]
},
ReturnConsumedCapacity='TOTAL',
ReturnItemCollectionMetrics='SIZE',
)
```

```
print("BatchputItem succ!")
print(json.dumps(response, indent=4))

#update data
t_name='music'
table = dynamodb.Table(t_name)

response = table.update_item(
    Key={
        "Artist": "none know",
        "SongTitle": "test World"
    },
    UpdateExpression="SET test = :incr REMOVE Tags.del",
    ExpressionAttributeValues={
        ":incr": 'cat'
    },
    ReturnValues="UPDATED_NEW"
)

print("updateItem succ")
print(json.dumps(response, indent=4))

#query data
t_name='music'
table = dynamodb.Table(t_name)

response = table.query(
    KeyConditionExpression=Key('SongTitle').eq('test World')
)

items = response['Items']
print(items)

#delete Item
test_table='music'
table = dynamodb.Table(t_name)
response = table.delete_item(
    Key={
        'Artist': 'none know',
        "SongTitle": "test World",
```

```
}  
)  
  
print("DeleteItem succ!")  
print(json.dumps(response, indent=4))
```

运行结果

```
[root@TENCENT64 /data/test/dynamodb]#  
[root@TENCENT64 /data/test/dynamodb]# python test.py  
Table status: dynamodb.Table(name='music')  
BatchputItem succ!  
{  
  "ResponseMetadata": {  
    "HTTPStatusCode": 200,  
    "RequestId": "635655159810"  
  }  
}  
updateItem succ  
{  
  "Attributes": {  
    "test": "cat"  
  },  
  "ResponseMetadata": {  
    "HTTPStatusCode": 200,  
    "RequestId": "635655159811"  
  }  
}  
[{"AlbumTitle": u'The Buck Starts Here', u'Price': Decimal('0.99'), u'Artist': u'none know', u'Genre': u'Rock', u'test': u'cat', u'SongTitle': u'test world'}]  
DeleteItem succ!  
{  
  "ResponseMetadata": {  
    "HTTPStatusCode": 200,  
    "RequestId": "635655159813"  
  }  
}  
[root@TENCENT64 /data/test/dynamodb]#
```


重连机制

最近更新时间：2018-09-14 16:27:06

说明

腾讯云MongoDB数据库服务提供的不是简单的mongod访问，给到用户访问的是一个负载均衡IP，此IP后面是连接到一系列类似mongos一样存在的路由接入层。

客户端驱动会透过负载均衡IP与接入机建立一个长连接，当此连接处于长期间活跃状态时，我们不会对其做任何干预，但是当长连接闲置时间超过1天时（此时间会随着版本优化而调整），路由接入层会踢掉该连接。

一般来说，客户端驱动会实现一个自动重连的过程，但是也有部分语言的驱动并没有实现。对于没有实现自动重连的语言驱动，当用户使用一个已经被踢掉的连接来尝试与腾讯云MongoDB服务通信时可能会得到“Remote server has closed the connection”之类的错误信息。所以得手动进行重连，这里给出一个PHP重连的demo。

基于php mongo驱动的重连实现

```
<?php

function getConnection() {
    $connection = false;
    $uri = 'mongodb://rwuser:1234567a@10.66.148.142:27017/admin?authMechanism=MONGODB-CR';
    $maxRetries = 5;
    for( $counts = 1; $counts <= $maxRetries; $counts++ ) {
        try {
            $connection = new MongoClient($uri);
        } catch( Exception $e ) {
            // 或者 根据需要使用下面的catch代码行，注意那一个“\”，某些框架使用命名空间时需要用到。
            // } catch( \Exception $e ) {
            continue;
        }
        break;
    }
    return $connection;
}

$connection = getConnection();

if($connection) {
    $db = $connection->testdb;
    $collection = $db->testcollection;
```

```
$one = $collection->findOne();  
  
var_dump($one);  
}
```

导入导出

最近更新时间：2018-09-14 16:27:40

在腾讯云云服务器（CVM）中可用MongoDB提供的shell客户端连接腾讯云MongoDB服务进行数据导入和导出，请注意使用最新版本的MongoDB客户端套件，具体请参考【最佳实践】=>【[连接示例](#)】。

mongodump 和 mongorestore

MongoDB官方提供了两套数据导入导出工具，一般来说，进行整库导出导入时使用[mongodump](#)和[mongorestore](#)，这一对组合操作的数据是BSON格式，进行大量dump和restore时效率较高。

典型的导出命令如下：

```
mongodump --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --db=testdb -o /data/dump_testdb
```

如图：

```
#: ./mongodump --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --db=testdb -o /data/dump_testdb
2016-11-16T12:12:06.114+0800    writing testdb.system.indexes to
2016-11-16T12:12:06.116+0800    done dumping testdb.system.indexes (1 document)
2016-11-16T12:12:06.116+0800    writing testdb.testcollection to
2016-11-16T12:12:06.118+0800    done dumping testdb.testcollection (3 documents)
```

```
mongorestore --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --dir=/data/dump_testdb
```

```
#: ./mongorestore --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --dir=/data/dump_testdb
2016-11-16T12:13:23.654+0800    building a list of dbs and collections to restore from /data/dump_testdb dir
2016-11-16T12:13:23.678+0800    reading metadata for testdb.testcollection from /data/dump_testdb/testdb/testcollection.metadata.json
2016-11-16T12:13:23.678+0800    restoring testdb.testcollection from /data/dump_testdb/testdb/testcollection.bson
2016-11-16T12:13:23.740+0800    restoring indexes for collection testdb.testcollection from metadata
2016-11-16T12:13:23.740+0800    finished restoring testdb.testcollection (3 documents)
2016-11-16T12:13:23.741+0800    done
```

mongoexport 和 mongoimport

一般来说，进行单个集合导出导入时使用[mongoexport](#)和[mongoimport](#)，这一对组合操作的数据是JSON格式，可读性较高。

mongoexport示例：

```
mongoexport --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --db=testdb --collection=testcollection -o /data/export_testdb_testcollection.json
```

另外您也可以加上 `-f` 参数指定需要的字段，`-q` 参数指定一个查询条件来限定要导出的数据。

mongoimport示例：

```
mongoimport --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin --db=testdb --collection=testcollection2 --file=/data/export_testdb_testcollection.json
```

多种认证方式的参数说明

在[连接示例](#)一文有说明，腾讯云MongoDB默认提供了“rwuser”和“mongouser”两个用户名分别支持“MONGODB-CR”和“SCRAM-SHA-1”两种认证方式。

对于“mongouser”以及在控制台创建的所有新用户，在使用导出导入命令工具时根据上示例操作即可。

对于“rwuser”，需要在每个命令里加入参数“--authenticationMechanism=MONGODB-CR”。

用mongodump举例说明：

```
mongodump --host 10.66.187.127:27017 -u rwuser -p thepasswordA1 --authenticationDatabase=admin --authenticationMechanism=MONGODB-CR --db=testdb -o /data/dump_testdb
```