

# 时序数据库 CTSDB

## 产品简介

## 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

### 产品简介

产品概述

产品优势

产品性能

# 产品简介

## 产品概述

最近更新时间：2018-01-11 17:44:55

CTSDB 时序数据库是腾讯云推出的一款分布式、可扩展、支持近实时数据搜索与分析的时序数据库。该数据库能够提供高效读写、低成本存储、强大的聚合分析能力、实例监控以及数据查询结果可视化等功能。整个系统采用多节点多副本的部署方式，有效保证了数据的高可用性和安全性。CTSDB 在处理海量时序数据时有如下的突出优势：

- 高并发写入：数据先写入内存，再周期性的 Dump 为不可变的文件存储。且可以通过批量写入数据，降低网络开销。
- 低成本存储：通过数据上卷（Rollup），对历史数据做聚合，节省存储空间。同时利用合理的编码压缩算法，提高数据压缩比。
- 强大的聚合分析能力：支持丰富的聚合查询方式，不仅支持 avg、min、max 等常用的聚合方式，也支持 Group By、区间、Geo、嵌套等复杂聚合分析。

# 产品优势

最近更新时间：2018-01-11 17:44:59

## 高性能

支持批量写入、高并发查询，通过集群扩展，可线性提升系统性能。

## 易使用

丰富的数据类型，兼容Elasticsearch常用的API接口。控制台提供丰富的数据管理和运维功能，操作简单。

## 强大的聚合分析能力

支持max、min、avg、percentile、sum、count等常用聚合。和复杂的脚本聚合、时间区间聚合、GEO聚合和嵌套聚合等。

## 高可靠

支持多副本，分布式部署，数据自动均衡。

## 低成本

通过上卷表Rollup提高压缩比，降低存储成本。

# 产品性能

最近更新时间：2018-08-30 15:23:14

## 测试工具

[测试工具下载](#)

## 集群配置

存储容量 350GB，写入能力约5万点/秒。

## 数据样例

- tag：host，tag\_0，tag\_1，tag\_2，均为string类型
- field：value\_1，value\_2，均为float32类型
- 时间戳：timestamp，为unix\_millis类型

其中，tag中的host是必需的，其他tag的数量可通过测试脚本的 -tag 选项修改。

```
{
  "host": "host_24",
  "tag_0": "v_3411",
  "tag_1": "v_2656",
  "tag_2": "v_713",
  "value1": 7.967921,
  "value2": 2.0137544,
  "timestamp": 1515036902
}
```

## 测试步骤

### 1.建表

命令如下

```
#建立metric (注意这里tags里只有一个host, 但当写入的数据有新的字段时, 这些字段将自动作为tag, 放入tags里面)
curl -u {user}:{passwd} -XPUT {ctsdb_ip_port}/_metric/ctsdb_bench?pretty -d '{
"tags": {
"host": "string"
},
"fields":{
"value_1": "float",
"value_2": "float"
},
"time": {
"name": "timestamp",
"format": "epoch_second"
},
"options": {
"expire_day": -1,
"refresh_interval": "10s",
"number_of_shards": 3,
"number_of_replicas": 1,
"rolling_period": -1
}
}'
# 查询metric
curl {ctsdb_ip_port}/_metric/ctsdb_bench?pretty
# 删除metric (同时会删除对应的所有数据)
curl -XDELETE {ctsdb_ip_port}/_metric/ctsdb_bench
```

其中, {ctsdb\_ip\_port} 为腾讯云提供的CTSDB访问端口, {user}和{passwd}分别为用户名和密码。

## 2.写入数据

使用脚本批量写入, 参数简介:

脚本在附件中, ctsdb\_benchmark\bin 目录下

- batch int  
批量写入点数 (default 5000)
- print\_interval int  
打印写入速率的时间间隔, 单位: s (default 5)
- routing  
是否使用 routing 功能 (用于提高查询效率)

- run\_time int  
写入进程运行时间，单位：s (default 300)
- tag int  
每条数据Tag个数 (default 3)
- threads int  
写入并发数 (default 9)
- urls string  
CTSDb访问列表地址（格式：`http://{ip1}:{port1},http://{ip2}:{port2},http://{ip3}:{port3}`）  
-user\_passwd string  
用户名密码(格式：`{user}:{passwd}`)

使用样例：

```
./ctsd_bench -urls=http://10.125.32.102:9200,http://10.125.32.102:9201 -threads=10
```

脚本输出：

分为三部分，param：选项参数，qps：每段时间间隔内的 qps，result：最终结果

```
----- param -----  
tag: 3  
batch: 5000  
routing: false  
threads: 9  
urls: [***]  
print_interval: 10  
----- qps -----  
2018-01-04 12:31:27.889214892 +0800 CST , qps: 71500/s  
2018-01-04 12:31:37.889166997 +0800 CST , qps: 69500/s  
2018-01-04 12:31:47.889225539 +0800 CST , qps: 59500/s  
2018-01-04 12:31:57.889212547 +0800 CST , qps: 60000/s  
2018-01-04 12:32:07.88920677 +0800 CST , qps: 59000/s  
2018-01-04 12:32:17.889202309 +0800 CST , qps: 58000/s  
----- result -----  
doc count: 3775000, bulk count: 755, qps: 62916/s
```

可以看到峰值写入速率为7.1w/s，平均写入速率为6.3w/s

### 3.查看写入的数据量

命令如下

```
# 查询命令
curl '{ctsdb_ip_port}/_cat/indices/ctsdb_bench@0_-1?v'
#返回结果
health status indexuuid pri rep docs.count docs.deleted store.size pri.store.size
green open ctsdb_bench@0_-1 EKPS4Q9rRuuvoZYZ-wCQIQ 3 1 209700000 1.4gb733.1mb
```

#### 4.查询数据

样例查询 1：查询一段时间内某个 host的所有数据

```
# 查询命令
curl -XPOST "{ctsdb_ip_port}/ctsdb_bench/_search?pretty" -d '{
  "size": 100,
  "query": {
    "bool": {
      "filter": [
        {
          "range": {
            "timestamp": {
              "format": "yyy-MM-dd HH:mm:ss",
              "time_zone": "+08:00",
              "gte": "2018-01-01 00:00:00",
              "lt": "2018-01-08 00:00:00"
            }
          }
        }
      ],
      "term": {
        "host": "host_3368"
      }
    }
  },
  "docvalue_fields": [
    "host",
    "tag_0",
    "tag_1",
    "tag_2",
    "value_1",
    "value_2",
    "timestamp"
  ]
}'
```

```
# 返回结果
```

```
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 3,
    "successful" : 3,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 4194,
    "max_score" : 0.0,
    "hits" : [
      {
        "_index" : "ctsdb_bench@0_-1",
        "_type" : "doc",
        "_id" : "AWC_aZ3gxbEXnFyEZTmi",
        "_score" : 0.0,
        "fields" : {
          "host" : [
            "host_3368"
          ],
          "value_2" : [
            1651.0557861328125
          ],
          "tag_0" : [
            "v_266"
          ],
          "tag_1" : [
            "v_189"
          ],
          "tag_2" : [
            "v_4461"
          ],
          "value_1" : [
            3033.96337890625
          ],
          "timestamp" : [
            1515039821000
          ]
        }
      }
    ]
  }
}
```

```
"_type": "doc",
"_id": "AWC_aV_eMdRQ4gVPAh2d",
"_score": 0.0,
"fields": {
  "host": [
    "host_3368"
  ],
  "value_2": [
    3133.74658203125
  ],
  "tag_0": [
    "v_4505"
  ],
  "tag_1": [
    "v_1657"
  ],
  "tag_2": [
    "v_2986"
  ],
  "value_1": [
    535.3303833007812
  ],
  "timestamp": [
    1515039821000
  ]
}
},
{
  "_index": "ctsdb_bench@0_-1",
  "_type": "doc",
  "_id": "AWC_aWTdMdRQ4gVPAmu9",
  "_score": 0.0,
  "fields": {
    "host": [
      "host_3368"
    ],
    "value_2": [
      676.7848510742188
    ],
    "tag_0": [
      "v_3521"
    ],
    "tag_1": [
      "v_531"
    ],
  ],
}
```

```
"tag_2" : [
  "v_4813"
],
"value_1" : [
  243.2778778076172
],
"timestamp" : [
  1515039821000
]
}
}
.....
]
}
}
```

样例查询 2：查询一段时间内某个 host 的所有数据，对结果按分钟粒度分桶，计算每个桶内有多少条数据和桶内 value\_1 字段的最大值。

```
# 查询命令
curl -XPOST "{ctsdb_ip_port}/ctsdb_bench/_search?pretty" -d '{
  "size": 0,
  "query": {
    "bool": {
      "filter": [
        {
          "range": {
            "timestamp": {
              "format": "yyyy-MM-dd HH:mm:ss",
              "time_zone": "+08:00",
              "gte": "2018-01-01 00:00:00",
              "lt": "2018-01-08 00:00:00"
            }
          }
        }
      ],
      {
        "term": {
          "host": "host_3368"
        }
      }
    ]
  }
},
  "aggs": {
    "buckets": {
```

```
"date_histogram":{
  "field":"timestamp",
  "interval":"1m",
  "format":"yyyy-MM-dd-HH:mm:ss"
},
"aggs":{
  "max_value_1":{
    "max":{
      "field":"value_1"
    }
  }
}
}'
# 返回结果
{
  "took" : 73,
  "timed_out" : false,
  "_shards" : {
    "total" : 3,
    "successful" : 3,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 4194,
    "max_score" : 0.0,
    "hits" : [ ]
  },
  "aggregations" : {
    "buckets" : {
      "buckets" : [
        {
          "key_as_string" : "2018-01-04-04:17:00",
          "key" : 1515039420000,
          "doc_count" : 1300,
          "max_value_1" : {
            "value" : 3364.01318359375
          }
        },
        {
          "key_as_string" : "2018-01-04-04:18:00",
          "key" : 1515039480000,
          "doc_count" : 545,
```

```
"max_value_1" : {  
  "value" : 3364.951904296875  
}  
,  
{  
  "key_as_string" : "2018-01-04-04:19:00",  
  "key" : 1515039540000,  
  "doc_count" : 0,  
  "max_value_1" : {  
    "value" : null  
  }  
}  
.....  
]
```