

使用 TcaplusDB Client

TcaplusDB · 自研 NoSQL 数据库

Exported on 01/05/2021

Table of Contents

1	前言	6
2	下载和使用TcaplusDB Client.....	7
2.1	下载	7
2.2	安装	7
2.3	连接 TcaplusDB.....	8
3	TcaplusDB Client命令参考	10
3.1	[CLI] 启动参数	10
3.1.1	参数	10
3.1.2	查看业务ID和区域ID	11
3.1.2.1	进入业务管理->业务维护界面	11
3.1.2.2	点击具体业务	11
3.1.3	例子	12
3.1.4	TDR.....	12
3.1.4.1	tdr.....	12
3.1.4.2	tdr2xml	13
3.1.5	参考示例表结构	14
3.1.6	编码	14
3.2	[CLI]desc	14
3.2.1	简介	14
3.2.2	语法	14
3.2.3	参数	15
3.2.4	错误	15
3.2.5	例子	15
3.3	[CLI]insert	18
3.3.1	简介	19
3.3.2	语法	19
3.3.3	参数：	19
3.3.4	错误	20
3.3.5	例子	20
3.4	[CLI]select.....	21
3.4.1	简介	21

3.4.2	语法	21
3.4.3	参数	21
3.4.4	错误	22
3.4.5	例子	22
3.5	[CLI]select--索引查询	24
3.5.1	介绍	24
3.5.1.1	支持的sql查询语句	24
3.5.1.2	不支持的sql查询语句	28
3.6	[CLI]setttl	29
3.6.1	简介	29
3.6.2	语法	29
3.6.3	参数	29
3.6.4	错误	30
3.6.5	返回信息	30
3.6.6	例子	30
3.7	[CLI]getttl.....	30
3.7.1	简介	31
3.7.2	语法	31
3.7.3	参数	31
3.7.4	返回信息	31
3.7.5	错误	31
3.7.6	例子	32
3.8	[CLI]help	32
3.8.1	简介	32
3.8.2	语法	32
3.8.3	例子	32
3.9	[CLI]show.....	34
3.9.1	简介	34
3.9.2	语法	34
3.9.3	参数	34
3.9.4	例子	34
3.10	[CLI]count.....	35
3.10.1	简介	36
3.10.2	使用语法	36

3.10.3 错误	36
3.10.4 例子	36
3.11 [CLI]exit/quit	36
3.11.1 简介	36
3.11.2 使用语法	36
3.11.3 例子	37
3.12 [CLI]replace	37
3.12.1 简介	37
3.12.2 语法	37
3.12.3 参数	37
3.12.4 错误	38
3.12.5 例子	38
3.13 [CLI]update.....	39
3.13.1 简介	39
3.13.2 语法	39
3.13.3 参数	39
3.13.4 错误	40
3.13.5 例子	40
3.14 [CLI]delete	40
3.14.1 简介	40
3.14.2 语法	41
3.14.3 参数	41
3.14.4 错误	41
3.14.5 例子	41
3.15 [CLI]dump.....	42
3.15.1 简介	42
3.15.2 语法	42
3.15.3 参数	42
3.15.4 错误	43
3.15.5 例子	43
3.16 [CLI] load.....	44
3.16.1 简介	44
3.16.2 语法	44
3.16.3 参数	45

3.16.4 错误	45
3.16.5 例子	45
3.17 [CLI]常见错误返回.....	46

1 前言

TcaplusDB Client类似于MySQL的命令行工具，可方便用户对TcaplusDB表数据进行访问控制。Client提供类SQL语法的能力方便用户使用SQL语法进行表数据的操作，这个能力也是TcaplusDB相比其它NoSQL所特有的优势。

功能列表

Client支持的功能列表如下：

- 查看集群状态，如当前后端连接地址、访问ID、表格组ID
- 查看集群表信息，显示当前集群的表名列表
- 显示表定义详情
- 表数据操作，如插入、删除、查找、更新、替换
- 表行数统计
- 表数据dump, 支持dump到csv文件
- 表数据load, 支持从csv文件导入到表

使用限制

- 工具依赖CentOS系操作系统，建议CentOS7以上

命令列表

tcaplus_client支持的命令列表请参考：[TcaplusDB Client命令参考](#)(see page 6)。

2 下载和使用TcaplusDB Client

tcaplus_client 是一个 TcaplusDB 表访问的客户端工具，C++编写，可以用于从命令行管理TcaplusDB。目前工具集成在C++ SDK中。

2.1 下载

SDK	版本	下载地址
TcaplusServiceApi3.46.1.199000.x86_64_release_20201102.tar.gz	3.46.1	下载¹

2.2 安装

下载完成 TcaplusServiceApi 安装包后，将其上传至可与 TcaplusDB网络联通的服务器中。
上传完成后，执行下列命令解压安装包。

```
mkdir tcaplus && tar -xf TcaplusServiceApi3.46.1.199000.x86_64_release_20201102.tar.gz -C tcaplus
```

解压完成后，进入至 tcaplus 的 bin 目录中，并赋予可执行权限：

```
cd tcaplus/release/x86_64/bin chmod +x tcaplus_client
```

直接执行 ./tcaplus_client 命令，会提示连接数据库所需的参数信息，用户可以根据自己的集群信息进行填写。

注意：下文示例中，app_id 代表业务ID、zone_id 代表游戏区ID。

¹ https://tcaplus-tool-1302668961.cos.ap-shanghai.myqcloud.com/sdk/TcaplusServiceApi3.46.1.199000.x86_64_release_20201102.tar.gz

```
# ./tcaplus_client
-----
invalid parameters, please start the client as following:
./tcaplus_client -a access_id -z tablegroup_id [-s signature] -d dir_server_url [-t table_name] [-l log_file.xml] [-T tdr_file.tdr] [-e execute_command].
the params in [] are optional, and their order is not important.
-a(--app_id)    APP_ID or Access ID of Cluster.
-z(--zone_id)   ZONE ID or TableGroup Id.
-s(--signature) PASSWORD.
-d(--dir)       dir server addr.
-t(--table)      table to add.
-l(--log)        log file name that must be client_log.xml, and log class name be client.
-T(--tdr)        tdr filename.
-e(--execute)    SQL command need to execute, the content should be in quotes.
e.g. ./tcaplus_client -a 2 -z 3 -s "FE6533875C8385C3" -d 172.25.40.181:9999 -T table_test.tdr -e
"select a, b from table where key = 1;"
```

2.3 连接 TcaplusDB

使用命令连接 TcaplusDB，如下示例中，访问点信息如下：

- 业务ID：3（集群接入ID）
- 连接密码：xxx（集群/业务访问密码）
- 内网地址:内网端口：10.125.32.21:9999
- 游戏区ID：1（表格组id）

```
# ./tcaplus_client -a 2 -z 3 -s 51BD7E6215D6F0BE -d 9.135.8.93:9999

===== Welcome to use tcaplus_client, use "help" to show usage =====

tcaplus>
```

在提示符之后输入 help，可看到进一步的帮助信息，通过 > help 具体命令 可以查看具体使用方法。


```
tcapplus> help;
```

```
-----  
help: show usage of commands, example: "help select;".  
show: get server status related information. executing "help show;" for details.  
exit/quit: exit the client.  
count: print record number in the database.  
  
desc: print table field name and type.  
select: query records from database.  
insert: insert a new record into database.  
replace: replace a record into the database.  
update: update a record in the database.  
delete: delete record(s) from database.  
  
dump: dump records from database.  
load: load records into the database.  
-----
```

3 TcaplusDB Client命令参考

3.1 [CLI] 启动参数

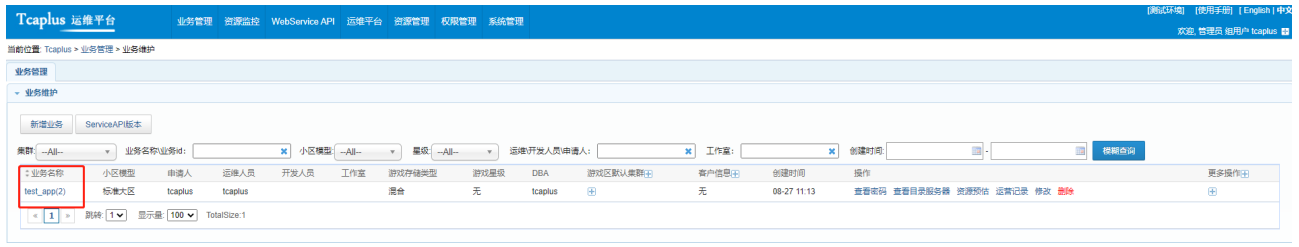
起始版本
3.46.0

3.1.1 参数

	docker环境说明	腾讯云环境说明	必填项
-a	业务ID	集群接入ID	是
-z	游戏区ID	表格组ID	是
-s	业务密码, 打开web运维平台, 查看业务密码	集群访问密码	是
-d	目录服务器ip地址及端口, ip:9999	集群连接地址及端口, ip:9999	是
-t	表格名	表格名	否
-l	日志级别设置, 文件名必须是client_log.xml	日志级别设置, 文件名必须是client_log.xml	否
-T	tdr文件路径	tdr文件路径	否
-e	需要执行的SQL语句	需要执行的SQL语句	否
-v	版本查询	版本查询	否
<	重定向SQL语句到client执行	重定向SQL语句到client执行	否

3.1.2 查看业务ID和区域ID

3.1.2.1 进入业务管理->业务维护界面



如图所示，2为业务ID。

3.1.2.2 点击具体业务



如图所示，3为游戏区ID。

3.1.3 例子

```
[root@yuxinfan-PC0 /mnt/e/client-test]# /mnt/e/tcaplus-copy/build64_release/tcaplus/src/client/
tcaplus_client -a 2 -z 3 -s C12901752D0D3347 -d 9.134.42.229:9999 -T /mnt/e/tdr/2.3.table_list.tdr

===== Welcome to use tcaplus_client, use "help" to show usage =====

tcaplus>
[root@yuxinfan-PC0 /mnt/e/client-test]# /mnt/e/tcaplus-copy/build64_release/tcaplus/src/client/
tcaplus_client -a 2 -z 3 -s C12901752D0D3347 -d 9.134.42.229:9999 -T /mnt/e/tdr/2.3.table_list.tdr -e "show
tables;"

-----
| Table Name          | Type      |
-----
| MTownRoleInfo       | GENERIC   |
| table_generic       | GENERIC   |
| table_generic_xiahuaxian | GENERIC   |
| table_list          | LIST      |
| test_table          | GENERIC   |
-----

[root@yuxinfan-PC0 /mnt/e/client-test]# /mnt/e/tcaplus-copy/build64_release/tcaplus/src/client/
tcaplus_client -a 2 -z 3 -s C12901752D0D3347 -d 9.134.42.229:9999 -T /mnt/e/tdr/2.3.table_list.tdr < /mnt/
e/clienttest/TDR-List-TableList.txt

[root@yuxinfan-PC0 /mnt/e/client-test]# /mnt/e/tcaplus-copy/build64_release/tcaplus/src/client/
tcaplus_client -a 2 -z 3 -d 9.134.42.229:9999
password:
===== Welcome to use tcaplus_client, use "help" to show usage =====

tcaplus>

[root@yuxinfan-PC0 /mnt/e/client-test]# /mnt/e/tcaplus-copy/build64_release/tcaplus/src/client/
tcaplus_client -v

===== tcaplus_client 2.0 version: 3.42.2.194287.x86_64_release_20190930 =====
```

3.1.4 TDR

3.1.4.1 tdr

如果需要使用using tdr，必须在client启动参数中添加tdr文件路径，可通过tdr工具(见下方示例文件)把多个XML格式的元数据描述库转换成二进制格式。如果多个XML文件之间存在依赖关系，则被依赖的XML文件必须放在参数表前面。

使用示例:

```

tdr -B -o ov_res.tdr ov_res.xml
    #xml格式元数据库生成.tdr二进制库
tdr -C -o ov_res.c --old_xml_tagset ov_res.xml
    #使用老标签集的xml格式元数据库生成.c文件
tdr -H -O "include" --add_custom_prefix="m_" --no_type_prefix
    #xml元数据库生成.h文件, 生成的文件保存在include目录,
    #结构体 (struct) /联合体 (union) 成员名添加前缀"m_", 但不添加类型前缀
tdr -G -m Pkg -x ATTR -o Pkg.xml net_protocol.xml
    #为Pkg生成xml格式的配置文件, 剪切版本为Pkg的最大版本, 文件名为Pkg.xml
tdr -T -u prefixfile
    #导出生成.h文件时使用的数据成员前缀表到文件prefixfile中
tdr -A --indent-size=8 net_protocol.xml
    #根据net_protocol.xml中描述的协议生成ActionScript3语言的类文件, 缩进大小为8个空格
tdr -P --indent-size=8 net_protocol.xml
    #根据net_protocol.xml中描述的协议生成C++语言的类文件, 缩进大小为8个空格
tdr -S --indent-size=8 net_protocol.xml
    #根据net_protocol.xml中描述的协议生成C#语言的类文件, 缩进大小为8个空格
tdr -E 0x83010404
    #查询错误号0x83010404对应的错误信息

```

示例文件：

[tdr²](#)

3.1.4.2 tdr2xml

tdr2xml工具（见下方示例文件）可以把二进制元数据文件反编译成一个Xml格式的元数据描述文件。

用法：

```

tdr2xml [-o --out_file=FILE] [-h --help] [-v --version] DRFILE
    各选项的含义如下：
-o, --out_file=FILE 指定输出文件的名字, 缺省为a.xml;
-h, --help 输出使用帮助;
-v, --version 输出版本信息

```

使用实例：

```
tdr2xml -o net_cs.xml net_cs.tdr
```

将保存在net_cs.tdr文件中的元数据描述二进制自定义格式转换成XML格式的描述文件。

示例文件：

[tdr2xml³](#)

TDR相关文档请参考 <http://ptp.oa.com/product/7>

² <https://wiki.woa.com/download/attachments/470040170/tdr?api=v2&modificationDate=1607479150000&version=1>

³ <https://wiki.woa.com/download/attachments/470040170/tdr2xml?api=v2&modificationDate=1607479151000&version=1>

3.1.5 参考示例表结构

为了方便说明命令字的使用，我们在示例中会使用到TDR表table_generic_xiahuaxian和table_list，它们的表结构可以通过下方链接下载。

[sim_comm.xml](#)⁴ [simtable_256KB.xml](#)⁵

3.1.6 编码

为了可视化数据，我们对以下几种类型的数据进行编码输出。

	PB	TDR
对象	json	十六进制
Byte	十六进制	十六进制
数组	json	十六进制

3.2 [CLI]desc

起始版本
3.20.0

3.2.1 简介

查看表的定义信息。嵌套字段只能看到其属性为嵌套类型，但是无法查看嵌套结构体的定义。

3.2.2 语法

⁴ https://wiki.woa.com/download/attachments/470040170/sim_comm.xml?api=v2&modificationDate=1607479151000&version=1

⁵ https://wiki.woa.com/download/attachments/470040170/simtable_256KB.xml?api=v2&modificationDate=1607479151000&version=1

```
desc [table]/[table.field] [using tdr];
```

3.2.3 参数

	PB	TDR
table	表格的名字	
table.field	不支持	显示嵌套字段的结构，必须配合using tdr使用。
using tdr	不支持	从tdr文件中读取表结构，该操作必须在启动client时提供tdr文件。

3.2.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.2.5 例子

```
tcaplus> desc table_list;
```

Table Structure **From:** Tcaplus System

TableName: table_list

TableType: LIST, IdlType: TDRXML

SvrTdrCurrentVersion: 5

Field	Type	Key	Index	Size	Version	DefaultValue	Count	Select
uin	unsigned long long	PRI,SPLT	-	8	1	-	1	-
QQ								
name	string	PRI	-	640	1	-	1	-
Name								
key1	unsigned char	PRI	-	1	1	-	1	-
key4								
level	int	-	-	4	1	1	1	-
-								
count	unsigned char	-	-	1	1	0	1	-
-								
array_count	unsigned int	-	-	4	1	1	1	-
-								
items	unsigned long long[]	-	-	8	1	-	10	-
-								
c_int8	char	-	-	1	1	-1	1	-
-								
c_uint8	unsigned char	-	-	1	1	2	1	-
-								
c_int16	short	-	-	2	1	-3	1	-
-								
c_uint16	unsigned short	-	-	2	1	4	1	-
-								


```

+-----+-----+-----+-----+-----+-----+
+-----+-----+
|c_int32      |int      | -      | -      | 4      | 1      | -5      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_uint32     |unsigned int| -      | -      | 4      | 1      | 6      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_int64      |long long  | -      | -      | 8      | 1      | -7      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_uint64     |unsigned long long| -      | -      | 8      | 1      | -      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_float      |float      | -      | -      | 4      | 2      | 1.234568 | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_double     |double     | -      | -      | 8      | 3      | 9.876543 | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_string     |string     | -      | -      | 200     | 3      | -      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_string_128K|string     | -      | -      | 131072  | 1      | 123456789 | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_string_256K|string     | -      | -      | 262144  | 1      | 123456789 | 1      | -
| -          |
+-----+-----+
+-----+-----+
|c_binary     |char[]     | -      | -      | 1       | 1      | -      | 10     | -
| -          |
+-----+-----+
+-----+-----+
|binary      |char       | -      | -      | 1       | 5      | -      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|selector     |short      | -      | -      | 2       | 1      | -      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|single_struct|struct_type| -      | -      | 119     | 1      | -      | 1      | -
| -          |
+-----+-----+
+-----+-----+
|simple_struct |simple_struct| -      | -      | 30      | 4      | -      | 1      | -
| -          |

```

```

+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+
|single_union_selector|char          | -      | -      |1      |1      |0      |1      | -
| -                  |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|single_union          |union_type      | -      | -      |64     |1      | -      |1      |
single_union_selector|-          |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|array                |TableInfo[]     | -      | -      |125    |1      | -      |3      | -
|三层嵌套struct|
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|c_union              |union_type      | -      | -      |64     |1      | -      |1      |selector
| -                  |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|union_array          |union_type[]    | -      | -      |64     |1      | -      |3      |selector
| -                  |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|c_struct            |struct_type     | -      | -      |119    |1      | -      |1      | -
| -                  |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
|struct_array         |struct_type[]   | -      | -      |119    |1      | -      |3      | -
| -                  |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+

```

32 rows in set (36252 us)

tcapplus> desc table_list.single_struct using tdr;

Table Structure From: Local tdr file(/mnt/e/tdr/2.3.table_list.tdr)

table_list.single_struct

---single_struct(struct)

| |---x(int)

| |---y(int)

| |---score(double)

| |---rank(uchar)

| |---title(string)

| |---level2_struct(struct)

| | |---uin(int)

| | |---name(string)

3.3 [CLI]insert

起始版本

3.20.0

3.3.1 简介

向表格中插入一条数据，可采用显示声明参数或传入文件两种方式操作。

3.3.2 语法

```
##显示声明参数的值插入一条数据
insert into table (key1, key2, value1, vlaue2) values (1, "abc", 2, "def") [after -1] [shift none/head/tail];
##从csv格式文件中读取并插入一条数据
insert into table infile result.csv [after -1] [shift none/head/tail];
##从xml格式文件中读取并插入一条数据，该操作必须在启动client时提供tdr文件
insert into table infile result.xml [after -1] [shift none/head/tail] using tdr;
```

3.3.3 参数：

参数	PB	TDR	必填
table	表格的名字		是
key	主键字段名		是
value	非主键字段名		至少一个或*
after	LIST表： n>0 表示从第n条数据插入； n=-2 表示从队首插入数据； n=-1 表示从队尾插入数据； n<-2 不支持		否
	GENERIC表： 不支持		\

参数	PB	TDR	必填
shift	如果表格大小超过阈值（即表格的最大size），可选择： none：不淘汰数据 head：从队头淘汰 tail：从队尾淘汰数据		否
using tdr	不支持	以xml格式插入数据，文件结构必须严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
infile	从文件中读取数据		否

3.3.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.3.5 例子

```
tcaplus>insert into game_players (player_id,player_name,player_email,game_server_id) values (2,name,email,2);
insert success

insert time: 45322 us

tcaplus> Insert into table_list (uin, name, key1) values (99,99,99) after -1 shift tail;

insert success

insert time: 22464 us

tcaplus> Insert into table_list infile result.xml using tdr;

insert success

insert time: 9493 us

tcaplus> Insert into table_list infile result.csv;

insert success

insert time: 22368 us
```

示例文件：

[result.xml](#)⁶ [result.csv](#)⁷

3.4 [CLI]select

起始版本
3.20.0

3.4.1 简介

用户可以使用select命令从数据库中获取指定的整条记录的值或部分记录的部分字段的值，如果没有匹配的记录，将会返回错误。3.46.0以后的版本将支持[分布式索引查询](#)⁸。

3.4.2 语法

```
select key1, key2, key3, value1, value2 [into result.csv] from table where key1 = 1 and key2 = "abc" [and
-index = 1] [\P] [\G] [using tdr]
select * [into result.xml] from table where key1 = 1 and key2 = "abc" [and -index = 1] using tdr [\P];
```

3.4.3 参数

	PB	TDR	必填项
table	表格的名字		是
key	主键字段名，支持分布式索引查询，可填入部分key值	主键字段名，必须填入所有key值	是
value	非主键字段名		至少一个或*
-index	LIST表：如果指定“-index”会返回相同key下的第index条记录，如果不指定“-index”，则返回所有记录		否
	GENERIC表：不支持		\

⁶ <https://wiki.woa.com/download/attachments/470040183/result.xml?api=v2&modificationDate=1607479152000&version=1>

⁷ <https://wiki.woa.com/download/attachments/470040183/result.csv?api=v2&modificationDate=1607479152000&version=1>

⁸ <https://wiki.woa.com/pages/viewpage.action?pageId=345187760&src=sidebar>

	PB	TDR	必填项
\P	打印时延数据		否
\G	竖排打印		否
using tdr	不支持	以xml格式输出数据，文件结构必须严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
into	输出数据到文件		否

3.4.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.4.5 例子

```

tcaplus> select * from test_table where gameid=1234 and itemid=12323 and name=testname;
+-----+-----+-----+-----+-----+
|gameid|itemid|name      |typeid|Data|uname|
+-----+-----+-----+-----+
|1234  |12323 |"testname"|0      |9   |"ab" |
+-----+-----+-----+-----+
1 records selectd, select time: 9802 us

tcaplus> select uname from test_table where gameid=1234 and itemid=12323 and name=testname;
+-----+-----+-----+
|gameid|itemid|name      |uname|
+-----+-----+-----+
|1234  |12323 |"testname"|"ab" |
+-----+-----+-----+
1 records selectd, select time: 9457 us

tcaplus> select * into test.txt from test_table where gameid=1234 and itemid=12323 and name=testname;
1 records are stored to test.csv, select time: 10198 us

tcaplus> select * from test_table where gameid=1234 and itemid=12323 and name=testname \P \G;
gameid: 1234
itemid: 12323
name: "testname"
typeid: 0
Data: 9
uname: "ab"

API ----      -1us      ---->ProxyFront-----      10us      ---->ProxyEnd---      364us      ---->SvrMainStart
|
|11380us      |4138us      |4104us      |381us
|
|61us
API <---      34197us      ---->ProxyFront<---      24us      ---->ProxyEnd<---      3298us      ---->SvrWorkerEnd
1 records selectd, select time: 11380 us

tcaplus> select * into table_list.xml from table_list where uin=99 and name = "99" and key1=99 using tdr;
11 records are stored to table_list.xml, select time: 135299 us

tcaplus> select c_string from table_list where uin=99 and name = "99" and key1=99;
+---+---+---+---+---+
|uin|name|key1|c_string|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
|99|"99"|99|""|
+---+---+---+---+---+
|99|"99"|99|""|
+---+---+---+---+---+
|99|"99"|99|""|
+---+---+---+---+---+
|99|"99"|99|""|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
|99|"99"|99|" "|

```

```

+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
11 records selectd, select time: 102572 us

tcaplus> select c_string from table_list where uin=99 and name = "99" and key1=99 and -index=9;
+---+---+---+---+---+
|uin|name|key1|c_string|
+---+---+---+---+---+
|99|"99"|99|" "|
+---+---+---+---+---+
1 records selectd, select time: 9886 us

```

3.5 [CLI]select--索引查询

起始版本
3.46.0

3.5.1 介绍

TcaplusDB提供全局索引的能力，以sql查询语句进行索引查询，其中，sql查询条件中的字段必须是建立了全局索引的字段，另外，如果是聚合查询，那么聚合查询的字段也必须是建立了全局索引的字段；

一个索引查询请求，最多返回3000条记录；

3.5.1.1 支持的sql查询语句

条件查询

支持 =, >, >=, <, <=, !=, between, in, not in, like, not like, and, or , 比如:


```

tcaplus> select * from pb_generic_index_shardingkey where openid>10 and tconndid<1000;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|openid|timekey |tconndid|svrid  |gamesvrid |other_property |items|lockid |
|pay|id_uint32|id_int32|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|201   |"timekey"|201     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|200   |"timekey"|200     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|203   |"timekey"|203     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|204   |"timekey"|204     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|202   |"timekey"|202     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+

total 5 records

tcaplus> select * from pb_generic_index_shardingkey where openid between 1 and 300 and tconndid<1000;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|openid|timekey |tconndid|svrid  |gamesvrid |other_property |items|lockid |
|pay|id_uint32|id_int32|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|203   |"timekey"|203     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|204   |"timekey"|204     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|201   |"timekey"|201     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|200   |"timekey"|200     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+

```

```

|202  |"timekey"|202      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+

total 5 records

tcaplus> select * from pb_generic_index_shardingkey where openid>10 or tconndid<1000;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|openid|timekey|tconndid|svrid  |gamesvrid  |other_property  |items|lockid  |
|pay|id_uint32|id_int32|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|202  |"timekey"|202      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|203  |"timekey"|203      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|201  |"timekey"|201      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|200  |"timekey"|200      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|204  |"timekey"|204      |"svrid"|"gamesvrid"| [{"key":1,"value":1},{ "key":2,"value":2}]|-  |[1,2,3,4]|-
|1      |1      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+

total 5 records

```

注意：between查询时，between a and b，对应的查询范围为[a, b]，比如 between 1 and 100, 是会包含1和100这两个值的，即查询范围为[1,100]；

注意：like查询是支持模糊匹配，其中"%"通配符，匹配0个或者多个字符；"_"通配符，匹配1个字符；

分页查询

支持limit offset分页查询，比如：

```

tcaplus> select * from pb_generic_index_shardingkey where openid>10 limit 3 offset 0;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|openid|timekey |tconndid|svrid  |gamesvrid |other_property |items|lockid  |
|pay|id_uint32|id_int32|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|200   |"timekey"|200     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |         |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|201   |"timekey"|201     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |         |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|202   |"timekey"|202     |"svrid"|"gamesvrid"| [{"key":1,"value":1}, {"key":2,"value":2}]|- | [1,2,3,4] |-
|1     |1        |         |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+

```

注意：当前limit必须与offset搭配使用，即不支持limit 1 或者 limit 0,1这种；

聚合查询

当前支持的聚合查询包括：sum, count, max, min, avg, 比如：

```

tcaplus> select sum(openid), count(*), max(openid), avg(openid) from pb_generic_index_shardingkey where
  openid>10 ;
1010,5,204,202

```

注意：聚合查询不支持limit offset，即limit offset 不生效；

注意：目前只有count支持distinct，即 select count(distinct(a)) from table where a > 1000; 其他情况均不支持 distinct

部分字段查询

支持查询部分字段的值，比如：

```

tcaplus> select svrid,gamesvrid from pb_generic_index_shardingkey where openid>10 or tconndid<1000;
+-----+-----+-----+-----+-----+
|openid|timekey |tconndid|svrid  |gamesvrid |
+-----+-----+-----+-----+-----+
|204   |"timekey"|204     |"svrid"|"gamesvrid"|
+-----+-----+-----+-----+-----+
|203   |"timekey"|203     |"svrid"|"gamesvrid"|
+-----+-----+-----+-----+-----+
|202   |"timekey"|202     |"svrid"|"gamesvrid"|
+-----+-----+-----+-----+-----+
|200   |"timekey"|200     |"svrid"|"gamesvrid"|
+-----+-----+-----+-----+-----+
|201   |"timekey"|201     |"svrid"|"gamesvrid"|
+-----+-----+-----+-----+-----+

total 5 records

```

对于pb表，还支持查询嵌套字段的值，类似：select field1.field2.field3, a, b from table where a > 1000;

3.5.1.2 不支持的sql查询语句

不支持聚合查询与非聚合查询混用

select *, a, b from table where a > 1000;

select sum(a), a, b from table where a > 1000;

select count(*), * from table where a > 1000;

不支持order by查询

select * from table where a > 1000 limit 100 offset 0;

不支持group by查询

select * from table where a > 1000 group by a;

不支持having查询

select sum(a) from table where a > 1000 group by a having sum(a) > 10000;

不支持多表联合查询

select * from table1 where table1.a > 1000 and table1.a = table2.b;

不支持嵌套select查询

select * from table where a > 1000 and b in (select b from table where b < 5000);

不支持别名

select sum(a) as sum_a from table where a > 1000;

不支持的其他查询

- (1) 不支持join查询；
- (2) 不支持union查询；
- (3) 不支持类似 select a+b from table where a > 1000 的查询；
- (4) 不支持类似 select * from table where a+b > 1000 的查询；
- (5) 不支持类似 select * from table where a >= b 的查询；
- (6) 不支持其他未提到的查询；

3.6 [CLI]setttl

起始版本
3.50.0

3.6.1 简介

用户可以通过调用SetTTL命令，为记录设置一个生存时间（Time To Live,TTL）：记录的生存时间（精度为毫秒）在设置之后就会随着时间的流逝而不断地减少，当一个记录的生存时间被消耗殆尽时，TcaplusDB就会移除这个记录。SetTTL命令只针对单条记录生效，不支持针对多条记录的操作。

3.6.2 语法

```
setttl [table] ttl=[TTL] where key1 = 1 and key2 = "abc";
```

3.6.3 参数

参数	是否必填	限制条件	说明
table	是	无	表名
TTL	是	最大不能超过uint64_t最大值的一半，即ttl最大值为ULONG_MAX/2，超过该值接口会强制设置为该值	生存时间，以毫秒为单位
where条件中的key	是	对于TDR表，必须填入所有key值	声明key的值，多个key值用and连接

3.6.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.6.5 返回信息

情况描述	返回信息
设置成功	Set time to live successfully.
记录不存在或者已过期	Record does not exist or has expired.
设置失败	Failed to set time to live. The error code is [error code] and the error message is [Error message].

3.6.6 例子

设置2000毫秒的生存时间

```
tcaplus> setttl mails ttl=2000 where key1 = 1 and key2 = "abc";
Set time to live successfully.
```

3.7 [CLI]getttl

起始版本
3.50.0

3.7.1 简介

用户可以通过调用GetTTL命令，获取记录的生存时间（Time To Live,TTL）：在为记录设置了生存时间之后，用户可以使用GetTTL命令查看键的剩余生存时间（精度为毫秒），即键还有多久才会因为过期而被移除。GetTTL命令只针对单条记录生效，不支持针对多条记录的操作。

3.7.2 语法

```
getttl from [table] where key1 = 1 and key2 = "abc";
```

3.7.3 参数

参数	是否必填	限制条件	说明
table	是	无	表名
where条件中的key	是	对于TDR表，必须填入所有key值	声明key的值，多个key值用and连接

3.7.4 返回信息

情况描述	返回信息
获取成功	The time to live is [TTL] milliseconds.
keys不存在或者已过期	Record does not exist or has expired.
keys存在并且没有设置过期时间（永久有效）	Record exists and no expiration time is set (permanent).
获取失败	Failed to get time to live. The error code is [error code] and the error message is [Error message].

3.7.5 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.7.6 例子

获取某条记录已经设置的生存时间

```
tcaplus> getttl from mails where key1 = 1 and key2 = "abc";  
The time to live is 2000 milliseconds.
```

3.8 [CLI]help

起始版本
3.20.0

3.8.1 简介

获取指令使用方法。

3.8.2 语法

```
##获取所有指令及其说明  
help;  
  
##获取指定指令名的详细说明  
help [指令名];
```

3.8.3 例子


```
tcaplus> help;
```

```
-----
help: show usage of commands, example: "help select;".
show: get server status related information. executing "help show;" for details.
exit/quit: exit the client.
```

```
count: print record number in the database.
```

```
desc: print table field name and type.
```

```
select: query records from database.
```

```
insert: insert a new record into database.
```

```
replace: replace a record into the database.
```

```
update: update a record in the database.
```

```
delete: delete record(s) from database.
```

```
dump: dump records from database.
```

```
load: load records into the database.
```

```
setttl: set ttl for a record
```

```
getttl: get ttl for a record
```

```
tcaplus> help select;
```

```
-----
example: select key1, key2, key3, value1, value2 [into result.csv] from table where key1 = 1 and key2 = "abc" [and -index = 1] [\P] [\G];
```

query records from database, you can specify part of the fields or whole fields (select *), and you can write the result to a file, which can be used by "insert" and "load"

for generic table, if the key in where clause is not complete, then it will send "GetByPartkey"

for list table, if "-index" is not specified in where clause, then it will send "ListGetAll",

otherwise it will send "ListGet"

\P: print time usage in detail

\G: print fields in column

Note: "-index" only used for list table

```
example: select * [into result.xml] from table where key1 = 1 and key2 = "abc" [and -index = 1] using tdr [\P];
```

if you specify "using tdr", then the records will be parsed by tdr file and print in xml format.

you can write the result into a file, which can be used by "load"

it only support "select *" instead of select part of the fields when specify "using tdr"

Note: "-index" only used for list table

globe index query:

```
example: select * from table where key1 > 1 and value1 > 100;
```

```
example: select * from table where value1 like "test";
```

```
example: select field1, field2 from table where key1 > 1 or value1 > 100;
```

```
example: select * from table where value1 between 100 and 200;
```

```
example: select * from table where value1 > 100 limit 100 offset 0;
```

```
example: select sum(value2), max(value2), min(value2), avg(value2), count(*) from table where value1 > 100;
```

Note: globe index query is only support generic table;

Note: current support: =, !=, >, >=, <, <=, like, not like, between, in, not in, and, or, limit offset;

Note: current support aggregation: count, sum, max, min, avg;

Note: for protobuf table, it support: "select field1.field2 from test where value1 > 100";

Note: limit must be used with offset, lack offset will query failed;

```
Note: the fields in where condition and in aggregation must had already created index;
Note: it not support: store the result to a file, such as "select * into file XXX" is not support;
Note: it not support: "select * from table"; which means to traverse table, you can used api traverse
method to traverse table;
Note: it not support: order by, group by, having, join, union and so on;
Note: it not support: select a+b XXX; select * from table where a+b>0; select sum(XX),field1 from XXX;
select *,field1 from XXX; .....
```

3.9 [CLI]show

起始版本
3.46.0

3.9.1 简介

查询服务端/表格基本信息。show tables可查询表格类型，协议类型。show status可查询当前的连接状态，目录服务器信息以及接入层信息。

3.9.2 语法

```
show [status/tables];
```

3.9.3 参数

	说明
status	查询当前服务端状态及配置信息
tables	查询所有表格信息

3.9.4 例子

```
tcaplus> show tables;
```

Table Name	Type	Protocol
test_table	GENERIC	TDR
tbMailTest	LIST	PROTOBUF
pb_generic_index_shardingkey	GENERIC	PROTOBUF
pb_generic_index_noshardkey	GENERIC	PROTOBUF
pb_generic_noindex_noshardkey	GENERIC	PROTOBUF
pb_list	LIST	PROTOBUF
pb_list2	LIST	PROTOBUF
pb_sortedlist	LIST	PROTOBUF
aes_info	GENERIC	TDR
auth_info	GENERIC	TDR
depend_me_services	GENERIC	TDR
host_info	GENERIC	TDR
instance_info	GENERIC	TDR
node_info	GENERIC	TDR
service_depends	GENERIC	TDR
service_info	GENERIC	TDR
token_info	GENERIC	TDR
cl_list	LIST	PROTOBUF
cl_generic	GENERIC	PROTOBUF
table_generic	GENERIC	TDR

```
tcaplus> show status;
```

```
Access_id(app_id) = 2
```

```
TableGroup_id(zone_id) = 3
```

```
[dir_server_url]
    dir_server_url[0]: 9.135.8.93:9999
```

```
[Proxys]
    proxy[0]:tcp://9.135.8.93:13755 CONNECTED Authened
    proxy[1]:tcp://9.135.8.93:13756 CONNECTED Authened
```

```
[The connected dir server]
    tcp://9.135.8.93:9999
```

3.10 [CLI]count

起始版本
3.20.0

3.10.1 简介

查询表格的记录总数

3.10.2 使用语法

```
count [表格名];
```

3.10.3 错误

参考 [\[CLI\]常见错误返回](#)(see page 46)

3.10.4 例子

```
tcaplus> count table_list;
-----
| TableName                Count                |
-----
| table_list                16                    |
-----
```

3.11 [CLI]exit/quit

起始版本
3.20.0

3.11.1 简介

退出程序

3.11.2 使用语法

```
exit/quit
```

3.11.3 例子

```
tcaplus> exit
tcaplus> quit
```

3.12 [CLI]replace

起始版本
3.20.0

3.12.1 简介

替换指定记录，如果记录不存在则新增记录，如果记录存在则更新记录。

3.12.2 语法

```
##显示声明字段的值，替换对应的记录
replace into table (key1, key2, value1, vlaue2, value3) values (1, "abc", 2, "def", 0x123456, 1) [where
-index = 1];

##导入csv文件替换对应的记录
replace into table infile 文件名 [where -index = 1];

##导入xml文件替换对应的记录
replace into table infile 文件名 [where -index = 1] using tdr;
```

3.12.3 参数

参数	PB	TDR	必填项
table	表格的名字		是
key	主键字段名，必须填入所有key值		是
value	非主键字段名		至少一个或*

参数	PB	TDR	必填项
-index	LIST表 ：必须指定 "-index", 只替换指定记录		是
	GENERIC表 ：不支持		\
using tdr	不支持	以xml格式替换数据，文件结构必须严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
infile	从文件中读取数据		否

3.12.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 38)

3.12.5 例子

```

tcaplus> replace into table_generic_xiahuaxian (_uin, name, _key3, key4, binary_count,
c_binary,c_binary1,c_binary2,c_binary3) values (99, "danmi_test_1",
4, 9, 3,0x64616e, 0x646162, 0x646162, 0x646162);

replace success

replace time: 42036 us

tcaplus> replace into table_list (uin, name, key1,level,count) values (99,99,99,22,11) where -index=1;

replace success

replace time: 10109 us

tcaplus> replace into table_list infile result.csv where -index=1;

replace success

replace time: 10054 us

tcaplus> replace into table_list infile result.xml where -index=1 using tdr;

replace success

replace time: 9836 us

```

示例文件：

[result.csv](#)⁹ [result.xml](#)¹⁰

3.13 [CLI]update

起始版本
3.20.0

3.13.1 简介

更新表格中一条数据，可采用显示声明参数或传入文件两种方式操作。

3.13.2 语法

```
##显示声明字段的值，更新对应的记录
update table set value1 = 1, value2 = "abc", value3 = 0x123456 where key1 = 1 and key2 = "abc" and [-index = 1];

##导入csv文件替换对应的记录
update table infile 文件名 [where -index = 0];

##导入xml文件替换对应的记录
update table infile 文件名 [where -index = 0] using tdr;
```

3.13.3 参数

参数	PB	TDR	必填项
table	表格的名字		是
key	主键字段名，必须填入所有key值		是
value	非主键字段名		至少一个或*

⁹ <https://iwiki.woa.com/download/attachments/470040197/result.csv?api=v2&modificationDate=1607479153000&version=1>

¹⁰ <https://iwiki.woa.com/download/attachments/470040197/result.xml?api=v2&modificationDate=1607479153000&version=1>

参数	PB	TDR	必填项
-index	LIST表：必须指定 "-index", 只替换指定记录		是
	GENERIC表：不支持		\
using tdr	不支持	以xml格式更新数据，文件结构必须严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
infile	从文件中读取数据		否
by partkey	不支持		\

3.13.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 40)

3.13.5 例子

```
tcaplus> update table_list set level=99 and count= 88 where uin=99 and name = "99" and key1=99 and -index=0
;

update success

update time: 117086 us
```

3.14 [CLI]delete

起始版本
3.20.0

3.14.1 简介

根据指定的key删除表格中一条数据，如果不指定-index则删除符合条件的所有记录。

3.14.2 语法

```
delete from table where key1 = 1 and key2 = "abc" [and -index = 1] [by partkey];
```

3.14.3 参数

参数	PB	TDR	必填项
table	表格的名字		是
key	主键字段名，必须填入所有key值		是
-index	LIST表 ：如果指定 "-index", 只删除指定记录；如果不指定"-index", 删除符合条件的所有记录。		是
	GENERIC表 ：不支持		\
by partkey	不支持	LIST表 不支持	\
		GENERIC表 通过表格的部分key删除记录	否

3.14.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 41)

3.14.5 例子

```
tcaplus> delete from table_list where uin=99 and name = "99" and key1=99 and -index=0;

delete success

delete time: 10263 us

tcaplus> delete from table_generic_xiahuaxian where _uin=99 and name = "danmi_test_1" and _key3=4 by
partkey;

delete success

delete time: 14405 us
```

3.15 [CLI]dump

起始版本
3.20.0

3.15.1 简介

全量导出表格中的数据，提供打印到控制台/输出到文件两种方式

3.15.2 语法

```
##导出部分字段
dump key1, key2, value1, value2 [into result.csv] from table limit 10;

##按xml格式导出
dump * [into 文件名] from table limit 10 using tdr;

##按csv格式导出
dump * [into 文件名] from table limit 10;
```

3.15.3 参数

参数	PB	TDR	必填项
table	表格的名字		是
key	主键字段名，必须填入所有key值		否
value	非主键字段名		否
limit	LIST表 ：导出key的个数，一个key对应多条记录		是
	GENERIC表 ：导出记录的条数，一个key对应一条记录		是
using tdr	不支持	以xml格式导出数据，文件结构严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
into	导出数据到文件		否

3.15.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 43)

3.15.5 例子

```

tcaplus> dump * from table_list limit 0;
uin, name, key1, level, count
, array_count, items, c_int8, c_uint8, c_int16, c_uint16, c_int32, c_uint32, c_int64, c_uint64, c_float, c_double, c_string, c_string_128K, c_string_256K, c_binary, binary
, selector, single_struct, simple_struct, single_union_selector, single_union, array, c_union, union_array, c_struct, struct_array
99, "99", 99, 1, 0, 1, 0x, -1, 2, -3, 4, -5, 6, -7, 0, 1.234568, 9.876543, "", "123456789", "123456789", 0x, 0, 0, 0x, 0x, 0, 0x, 0x, 0x, 0x, 0x, 0x, 0x
99, "99", 99, 1, 0, 1, 0x, -1, 2, -3, 4, -5, 6, -7, 0, 1.234568, 9.876543, "", "123456789", "123456789", 0x, 0, 0, 0x, 0x, 0, 0x, 0x, 0x, 0x, 0x, 0x, 0x
99, "99", 99, 1, 0, 1, 0x, -1, 2, -3, 4, -5, 6, -7, 0, 1.234568, 9.876543, "", "123456789", "123456789", 0x, 0, 0, 0x, 0x, 0, 0x, 0x, 0x, 0x, 0x, 0x, 0x
99, "99", 99, 1, 0, 1, 0x, -1, 2, -3, 4, -5, 6, -7, 0, 1.234568, 9.876543, "", "123456789", "123456789", 0x, 0, 0, 0x, 0x, 0, 0x, 0x, 0x, 0x, 0x, 0x, 0x

dump 4 records successful

dump time: 121671 us

tcaplus> dump * into table_list.txt from table_list limit 0;

dumped 4 records successful

tcaplus> dump * into table_list.xml from table_list limit 0 using tdr;

dumped 4 records successful

```

3.16 [CLI] load

起始版本
3.20.0

3.16.1 简介

导入数据，支持csv/xml两种数据格式，如果记录存在更新记录，如果记录不存在新增记录。

3.16.2 语法

```
##按xml格式导入
load table infile 文件名 using tdr;

##按csv格式导入
load table infile 文件名;
```

3.16.3 参数

参数	PB	TDR	必填项
table	表格的名字		是
using tdr	不支持	以xml格式导出数据，文件结构严格满足xml语法，该操作必须在启动client时提供tdr文件。	否
infile	从文件读取数据		是

3.16.4 错误

参考 [\[CLI\]常见错误返回](#)(see page 45)

3.16.5 例子

```
tcapplus> load table_list infile table_list_dump.xml using tdr;
loaded 49 records successful

tcapplus> load table_list infile table_list-dump.txt;
loaded 98 records successful
```

示例文件：

[table_list-dump.txt](#)¹¹ [table_list_dump.xml](#)¹²

¹¹ https://iwiki.woa.com/download/attachments/470040203/table_list-dump.txt?api=v2&modificationDate=1607479154000&version=1

¹² https://iwiki.woa.com/download/attachments/470040203/table_list_dump.xml?api=v2&modificationDate=1607479153000&version=1

3.17 [CLI]常见错误返回

错误返回	说明
cannot find key field	缺少主键或者无法识别主键字段名
cannot find value field	无法识别字段名
tcapsvr_fail_record_exist	当前记录已存在
table("table_name")registration failed	找到不到表格名为table_name的表格
proxy_err_query_for_convert_tcaplus_req_to_index_server_req_failed	没有设置索引服务