

DSDB C# SDK

by Dasudian

Contents

1 DSDB Csharp SDK	1
2 FAQ	2
3 更新日志	2
4 命名空间索引	2
4.1 包	2
5 类索引	2
5.1 类列表	2
6 文件索引	2
6.1 文件列表	2
7 命名空间文档	2
7.1 com 命名空间参考	3
7.2 com.dasudian 命名空间参考	3
7.3 com.dasudian.dsdb 命名空间参考	3
7.4 com.dasudian.dsdb.sdk 命名空间参考	3
7.4.1 枚举类型说明	3
8 类说明	4
8.1 DsdbClient类 参考	4
8.1.1 构造及析构造函数说明	4
8.1.2 成员函数说明	4
9 文件说明	6
9.1 DsdbClient.cs 文件参考	6
9.2 ErrorCode.cs 文件参考	7
9.3 FAQ.md 文件参考	7
9.4 release_note.txt 文件参考	7
索引	9

1 DSDB Csharp SDK

SDK提供了添加,删除,修改,查询数据的接口

如何使用SDK:

- 1: 创建一个客户端实例
- 2: 建立连接
- 3: 如果想添加数据,那么就使用put()
- 4: 或者使用get()得到具体某一条数据
- 5: 或者使用delete()删除具体某一条数据
- 6: 退出时,断开连接

2 FAQ

Q:

```
ErrorCode:CommunicationError  
ErrorMessage:Riak returned an error. Code '0'. Message: 'no_type', Unable to access a connection on the cluster (no more retries).
```

A:这是因为在服务器中并没有对应的bucketType, 请联系数据库管理员创建相应的bucketType

3 更新日志

Author	Date	Version	Note
Eden Wang	7/11/2017	1.1.0	发布第一个版本

4 命名空间索引

4.1 包

这里列出所有的包，附带简要说明(如果有的话):

com	3
com.dasudian	3
com.dasudian.dsdb	3
com.dasudian.dsdb.sdk	3

5 类索引

5.1 类列表

这里列出了所有类、结构、联合以及接口定义等，并附带简要说明:

DsdbClient	4
----------------------------	---

6 文件索引

6.1 文件列表

这里列出了所有文件，并附带简要说明:

DsdbClient.cs	6
ErrorCode.cs	7

7 命名空间文档

7.1 com 命名空间参考

命名空间

- namespace [dasudian](#)

7.2 com.dasudian 命名空间参考

命名空间

- namespace [dsdb](#)

7.3 com.dasudian.dsdb 命名空间参考

命名空间

- namespace [sdk](#)

7.4 com.dasudian.dsdb.sdk 命名空间参考

类

- class [DsdbClient](#)

枚举

- enum [ErrorCode](#) {
 [Success](#) = 0, [ShuttingDown](#) = -1, [NotFound](#) = -2, [CommunicationError](#) = -3,
 [InvalidResponse](#) = -4, [ClusterOffline](#) = -5, [NoConnections](#) = -6, [BatchException](#) = -7,
 [NoRetries](#) = -8, [HttpError](#) = -9, [InvalidRequest](#) = -10, [NoDsdbConfig](#) = -11,
 [IllegalParam](#) = -12 }

7.4.1 枚举类型说明

7.4.1.1 ErrorCode

```
enum ErrorCode [strong]
```

枚举值

Success	成功
ShuttingDown	服务器关闭
NotFound	未搜索到数据
CommunicationError	与数据库通讯失败
InvalidResponse	响应解析失败
ClusterOffline	集群不可达
NoConnections	未建立连接
BatchException	暂未使用
NoRetries	重试次数到达上限
HttpError	暂未使用
InvalidRequest	非法请求

枚举值

NoDsdbConfig	未找到配置项 dsdbConfig
IllegalParam	非法参数

8 类说明

8.1 DsdbClient 类 参考

Public 成员函数

- **DsdbClient** ()
构造函数
默认情况下 *debug* 选项设置为 *false*
- void **setDebug** ()
设置 *debug* 选项为 *true*
- **ErrorCode connect** ()
连接 *DSDB* 服务器
SDK 将从文件 *App.conf* 中读取 *configSection* 为 *dsdbConfig* 的配置项, 解析服务器地址并连接服务器
- void **disconnect** ()
断开与 *DSDB* 服务器的连接
- **ErrorCode put** (String bucketType, String bucket, String key, String value)
将数据存放在 *DSDB* 服务器中
在 *DSDB* 数据库中, 为了分类明确, 我们创建了三级索引.
一级索引为 *bucketType*, 二级索引为 *bucket*, 三级索引为 *key*, 其中 *<bucketType, bucket, key>* 联合唯一标识一条记录, 其值为 *value*
- **ErrorCode get** (String bucketType, String bucket, String key, out String value)
从数据库中取数据
在 *DSDB* 数据库中, 为了分类明确, 我们创建了三级索引.
一级索引为 *bucketType*, 二级索引为 *bucket*, 三级索引为 *key*, 其中 *<bucketType, bucket, key>* 联合唯一标识一条记录
- **ErrorCode delete** (String bucketType, String bucket, String key)
从数据库中删除数据
在 *DSDB* 数据库中, 为了分类明确, 我们创建了三级索引.
一级索引为 *bucketType*, 二级索引为 *bucket*, 三级索引为 *key*, 其中 *<bucketType, bucket, key>* 联合唯一标识一条记录

8.1.1 构造及析构函数说明

8.1.1.1 DsdbClient()

DsdbClient ()

构造函数

默认情况下 *debug* 选项设置为 *false*

8.1.2 成员函数说明

8.1.2.1 connect()

```
ErrorCode connect ( )
```

连接DSDB服务器

SDK将从文件App.conf中读取configSection为dsdbConfig的配置项, 解析服务器地址并连接服务器

返回

若ErrorCode.Success为成功, 其他错误码请参考ErrorCode

8.1.2.2 delete()

```
ErrorCode delete (
    String bucketType,
    String bucket,
    String key )
```

从数据库中删除数据

在DSDB数据库中, 为了分类明确, 我们创建了三级索引.

一级索引为bucketType, 二级索引为bucket, 三级索引为key, 其中 <bucketType, bucket, key>联合唯一标识一条记录

参数

<i>bucketType</i>	bucketType的名字, 必须由数据库管理员手动创建 并激活; 删除数据之前, 该bucketType应该已经被创建并被激活
<i>bucket</i>	bucket的名字
<i>key</i>	关键字的名字

返回

ErrorCode.Success为成功, 其他错误码请参考ErrorCode

8.1.2.3 disconnect()

```
void disconnect ( )
```

断开与DSDB服务器的连接

8.1.2.4 get()

```
ErrorCode get (
    String bucketType,
    String bucket,
    String key,
    out String value )
```

从数据库中取数据

在DSDB数据库中, 为了分类明确, 我们创建了三级索引.

一级索引为bucketType, 二级索引为bucket, 三级索引为key, 其中 <bucketType, bucket, key>联合唯一标识一条记录

参数

<i>bucketType</i>	bucketType的名字, 必须由数据库管理员手动创建 并激活; 查询数据之前, 该bucketType应该已经被创建并被激活
<i>bucket</i>	bucket的名字
<i>key</i>	关键字的名字
<i>value</i>	当返回值为 ErrorCode.Success , 返回对应的键值

返回

[ErrorCode.Success](#)为成功, 其他错误码请参考[ErrorCode](#)

8.1.2.5 put()

```
ErrorCode put (
    String bucketType,
    String bucket,
    String key,
    String value )
```

将数据存放在DSDB服务器中

在DSDB数据库中, 为了分类明确, 我们创建了三级索引.

一级索引为bucketType, 二级索引为bucket, 三级索引为key, 其中 <bucketType, bucket, key>联合唯一标识一条记录, 其值为value

参数

<i>bucketType</i>	bucketType的名字, 必须由数据库管理员手动创建并激活
<i>bucket</i>	bucket的名字
<i>key</i>	关键字的名字
<i>value</i>	值

返回

[ErrorCode.Success](#)为成功, 其他错误码请参考[ErrorCode](#)

8.1.2.6 setDebug()

```
void setDebug ( )
```

设置debug选项为true

该类的文档由以下文件生成:

- [DsdClient.cs](#)

9 文件说明

9.1 DsdClient.cs 文件参考

类

- class [DsdClient](#)

命名空间

- namespace [com.dasudian.dsdb.sdk](#)

9.2 ErrorCode.cs 文件参考

命名空间

- namespace [com.dasudian.dsdb.sdk](#)

枚举

- enum [ErrorCode](#) {
 [Success](#) = 0, [ShuttingDown](#) = -1, [NotFound](#) = -2, [CommunicationError](#) = -3,
 [InvalidResponse](#) = -4, [ClusterOffline](#) = -5, [NoConnections](#) = -6, [BatchException](#) = -7,
 [NoRetries](#) = -8, [HttpError](#) = -9, [InvalidRequest](#) = -10, [NoDsdbConfig](#) = -11,
 [IllegalParam](#) = -12 }

9.3 FAQ.md 文件参考

9.4 release_note.txt 文件参考

Index

- com, [3](#)
- com.dasudian, [3](#)
- com.dasudian.dsdb, [3](#)
- com.dasudian.dsdb.sdk, [3](#)
- com::dasudian::dsdb::sdk
 - ErrorCode, [3](#)
- com::dasudian::dsdb::sdk::DsdbClient
 - connect, [4](#)
 - delete, [5](#)
 - disconnect, [5](#)
 - DsdbClient, [4](#)
 - get, [5](#)
 - put, [6](#)
 - setDebug, [6](#)
- connect
 - com::dasudian::dsdb::sdk::DsdbClient, [4](#)
- delete
 - com::dasudian::dsdb::sdk::DsdbClient, [5](#)
- disconnect
 - com::dasudian::dsdb::sdk::DsdbClient, [5](#)
- DsdbClient, [4](#)
 - com::dasudian::dsdb::sdk::DsdbClient, [4](#)
- DsdbClient.cs, [6](#)
- ErrorCode
 - com::dasudian::dsdb::sdk, [3](#)
- ErrorCode.cs, [7](#)
- FAQ.md, [7](#)
- get
 - com::dasudian::dsdb::sdk::DsdbClient, [5](#)
- put
 - com::dasudian::dsdb::sdk::DsdbClient, [6](#)
- release_note.txt, [7](#)
- setDebug
 - com::dasudian::dsdb::sdk::DsdbClient, [6](#)