SensorDB V1.0.0系统使用手册

Version <1.0>

|  |  |
| --- | --- |
| 编制日期 | **2 2014年10月20日** |
| **编制作者** | **付俊伟** |
| **文档编号** |  |
| **文档类型** | **实验室运行文件** |

修订历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **描述** | **作者** |
| 2014年10月20日 | <1.0> | 文档创建 | 付俊伟 |
|  |  |  |  |

目 录

第1章 概述 4

§1.1 文档概述 4

§1.2 参考文档 4

§1.3 术语 4

第2章 配置 5

§2.1 配置说明 5

第3章 使用说明 7

§3.1 系统启动/停止 7

§3.1.1 SensorDB系统初始化： 7

§3.1.2 系统启动/停止 7

§3.2 CLI Shell 7

§3.2.1 登录 7

§3.2.2 表维护 7

§3.2.3 表数据维护 8

§3.2.4 系统状态查询 8

§3.2.5 命令帮助说明 8

§3.3 SensorDB 监控系统 9

第4章 编程说明 9

§4.1 Java API 9

§4.1.1 编程前准备 9

§4.1.2 连接SensorDB 9

§4.1.3 表操作 9

§4.1.4 表数据操作 10

§4.2 Python API 13

§4.2.1 安装 13

§4.2.2 连接SensorDB 13

§4.2.3 表操作 14

§4.2.4 表数据操作 15

# 概述

## 文档概述

SensorDB是提供传感器数据采集、存储、管理和查询的分布式数据库系统。本文档从配置和使用来介绍SensorDB系统，适合于系统管理员和程序开发者。

## 参考文档

## 术语

# 使用说明

## 系统启动/停止

### 系统启动/停止

#### 启动SensorDB

start-sensordb

#### 停止SensorDB

stop-sensordb

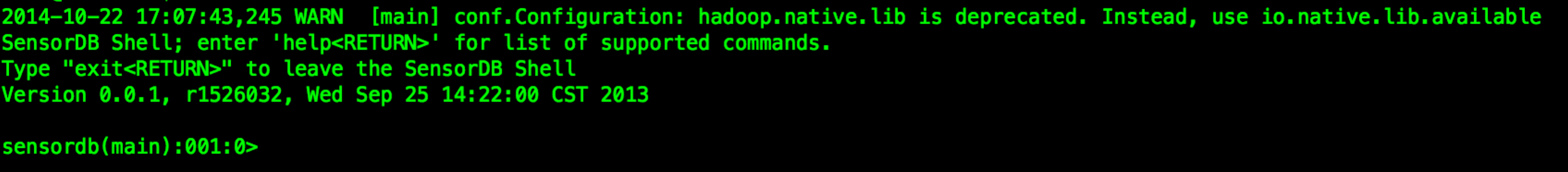
## CLI Shell

SensorDB 系统向用户提供了命令行工具方便进行数据库基本的查询维护。该命令行工具运行在Linux系统上。下面分别描述工具提供的各项命令及命令说明。

### 登录

请先ssh登录到SensorDB主机122.144.166.103，然后执行命令sensordb shell，即可进入SensorDB命令行交互程序。

cli会连接SensorDB，连接成功后,会有这样的提示：



### 表维护

表维护命令一共包括 5 种，命令说明如下表。

|  |  |  |
| --- | --- | --- |
| 命令 | 命令说明 | 示例 |
| list | 列出数据库中当前所有的表名 |  |
| describe ‘表名’ | 获取指定表的表信息。 | describe ‘t1’ |
| create ‘表名’,’列族名’ | 创建指定表，同时需要指定列族名称。 | describe ‘t1’,’cf’ |
| disable ‘表名’ | 设置指定表不可用。 | disable ‘t1’ |
| drop ‘表名’ | 删除指定表，删除表之前必须先执行disable命令。 | drop ‘t1’ |

### 表数据维护

表数据维护命令一共包括 种，命令说明如下表。

|  |  |  |
| --- | --- | --- |
| 命令 | 命令说明 | 示例 |
| put ‘表名’, ‘行健’, ‘列族名:列名’, ‘列值’,…, ‘列族名:列名’, ‘列值’ | 给指定表写入数据，写入数据时需要指定表名，行键，列名，列值。 | 往表t1中插入数据一行两列。  put ‘t1’, ‘r1’, ‘cf:c1’, ‘value1’, ‘cf:c2’, ‘value2’。 |
| get ‘表名’, ‘行健’, <‘列族名:列名’， … > | 获取指定表指定行的数据，如果不指定列名，则返回该行所有列数据。 | get ‘r1’;  get ‘r1’, ‘cf:c1’ |
| scan ‘表名’ | 扫描指定表，返回该表所有数据 | scan ‘t1’ |
| delete ‘表名’, ‘行健’, ‘列族名:列名’ | 删除指定表指定行列的数据 | delete ‘t1’, ‘cf:c1’ |

### 系统状态查询

命令status可以获取SensorDB 系统状态。系统状态有三种不同的粒度，分别是：summary, simple, detailed，默认是summary。在status命令后加上粒度就能获取对应的状态信息。

status

status ‘summary’

status ‘simple’

status ‘detailed’

### 命令帮助说明

如果对命令使用不熟悉，可以输入帮助命令获取该命令的使用说明。使用规则：

help ‘命令名’。比如：

获取get命令使用说明，则输入：help ‘get’。

## SensorDB 监控系统

SensorDB 系统提供了简易的Web监控系统，访问该系统可以了解SensorDB系统基本的使用情况，包括集群状态，数据库表及其数据分布等等

SensorDB Web监控系统访问地址：<http://[SensorDB> host ip]:60010/

# 编程说明

本部分向开发者描述了如何使用SensorDB 客户端API来进行数据库的各种操作。

SensorDB 提供了Java和Python两种语言的客户端API，Java API提供的功能比Python的要更多。

## Java API

### 编程前准备

使用该API编程进行SensorDB操作访问前，需要预先做一点准备：

* 解压sensordb-client.zip，将解压后的所有jar文件放在工程classpath下。

### 连接SensorDB

SensorDB Client API 提供类Connection作为对外的统一访问接口类。在进行各项操作之前，需要先成功连接上SensorDB。连接时需要提供SensorDB 主机名和端口，端口默认是6677。

编程示例：

Connection conn = **new** Connection(*HOST*, 6677);

conn.connect();

如果没有异常发生就表示连接成功，连接成功后就可以进行SensorDB 各项操作。

表操作结束后，需要关闭对SensorDB的连接，同时SensorDB释放由该连接创建的所有资源。

调用close()方法执行连接关闭。

### 表操作

SensorDB Client API的Connection类 提供三种表操作，包括创建表、删除表、查询表。在进行表操作之前需要先创建数据库连接。

#### 创建表

调用createTable(String tableName)，创建表，创建时只需指定表名。

#### 删除表

提供两种删除表的操作，删除指定表和删除符合条件的多张表。

* 删除指定表

调用dropTable(String tableName) ，删除指定表并删除该表的所有数据。

* 删除符合条件的多张表

调用dropTables(String regex)，删除数据中表名符合参数条件的所有表，表名满足正则匹配模式。

#### 查询表

提供三种表查询操作，查询数据库中所有表、查询数据库中符合条件的表，查询指定表是否存在。

* 查询数据库中所有表

调用listTableNames()， 返回数据库中所有表的表名

* 查询数据库中符合条件的表

调用listTableNames(String regex)， 返回数据库中满足指定条件的所有表的表名。

* 查询指定表是否存在

调用isExisted(String tableName)，返回指定表是否存在

### 表数据操作

SensorDB Client API的Connection类 提供三种表操作，包括插入表数据、查询表数据。在进行表操作之前需要先创建数据库连接。

#### 插入表数据

提供两种表数据插入方式，单行插入和多行批量插入。多行批量插入方式比单行插入方式显著提高插入性能，当需要插入数据行较多时，建议使用多行批量插入。

* 单行插入

调用put(String tableName, byte[] sensorID, long sampledTimestamp, double x, double y, double z, Map<String, bye[]> sampledValues) 执行单行数据插入。

在插入时需要提供如下参数：

* 表名： 插入数据库表表名；
* 传感器编号：传感器标识，字节数组类型，实际数据类型不限制，但都需要先转换为字节数组。在提供的API类库里，我们提供了Bytes类实现Java基本类型到byte数组的转换方法；
* 数据采样时间戳：数据采样时的时间戳信息，单位毫秒；
* 数据采样时空间位置信息：数据采样时传感器所在的空间位置，空间位置信息包括x轴，y轴，z轴位置。如果不能提供位置信息，可设置为0，缺点是无法进行空间数据搜索。
* 采样的数据集合：传感器采集到的各个指标数据，Map映射类型，key为指标名称，value为数据转换后的字节数组
* 多行批量插入

调用batchPut(String tableName, List<Item> items) 完成多行数据的批量插入。在插入前需要创建多行数据集合。

类Item代表每行数据，类Item结构定义描述：

class Item{

**public** ByteBuffer sensorID; // 传感器标识

**public** **long** sampledts; // 数据采样时间戳

**public** **double** x; // 数据采样时位置信息-x轴

**public** **double** y; // 数据采样时位置信息-y轴

**public** **double** z; // 数据采样时位置信息-z轴

**public** Map<String,ByteBuffer> values; // 采样数据集合

}

#### 查询表数据

提供三种不同的表数据查询方式，按传感器ID查询，按时间范围查询，按时间空间范围查询。

* 按传感器ID查询

当需要查询指定传感器ID在指定时间范围内采集的所有数据时，调用

get(String tableName, byte[] sensorID, String starttime, String endtime)，返回查询结果对象ResultSet类型。

get(String tableName, byte[] sensorID, Date starttime, Date endtime)，返回查询结果对象ResultSet类型。

以上两个方法按照单个传感器来查询。

gets(String tableName, byte[] sensorID, String starttime, String endtime)，返回查询结果对象ResultSet类型。

gets(String tableName, byte[] sensorID, Date starttime, Date endtime)，返回查询结果对象ResultSet类型。

以上两个方法可以查询多个。

调用时的注意事项：

* 传感器标识数据需要先转换为字节数组
* 如果参数starttime和endtime是字符串类型，则必须满足格式yyyy-MM-dd HH:mm:ss
* 按时间范围查询

当需要查找指定时间范围内的所有数据时，调用

get(String tableName, String starttime, String endtime)，返回查询结果对象ResultSet类型。

get(String tableName, Date starttime, Date endtime)，返回查询结果对象ResultSet类型。

调用时的注意事项：

* 如果参数starttime和endtime是字符串类型，则必须满足格式yyyy-MM-dd HH:mm:ss
* 按时间空间范围查询

当需要查找指定时间范围内的所有数据时，调用

get(String tableName, Location start, Location end, String starttime, String endtime)，返回查询结果对象ResultSet类型。

get(String tableName, Location start, Location end, Date starttime, Date endtime)，返回查询结果对象ResultSet类型。

调用时的注意事项：

* 如果参数starttime和endtime是字符串类型，则必须满足格式yyyy-MM-dd HH:mm:ss
* 参数start和end表示查询空间范围的开始位置和结束位置，Location对象类型。

#### 查询结果对象ResultSet使用说明

调用查询表数据方法都会返回ResultSet类对象，然后通过操作ResultSet类对象来完成结果数据的访问。

表查询方法无论是否存在数据都会返回该对象，所以通过查看ResultSet返回的结果码来判断本查询是否成功查询并返回数据。

getErrCode() 返回操作结果码，只有当返回0时表示查询找到符合条件的数据，其他码值说明无数据或查询时发生异常，各个错误码的信息可见ErrCode类。

getSize() 返回查询找到的记录数

当查询并返回数据后，可通过调用next() 方法来遍历结果集，并通过调用形如getX()的方法来获取对应列的数据，X为该列对应的数据类型。