目录

[OBD 缓存结构设计 1](#_Toc425794520)

[1.  概述 1](#_Toc425794521)

[1.1 目的 1](#_Toc425794522)

[1.2 适用范围 1](#_Toc425794523)

[2.  Redis简介 1](#_Toc425794524)

[2.1存储方式 1](#_Toc425794525)

[2.2 数据结构 2](#_Toc425794526)

[3.  OBD数据结构 5](#_Toc425794527)

[3.1服务程序运行时信息 5](#_Toc425794528)

[3.2 OBD设备链接信息 6](#_Toc425794529)

# OBD 缓存结构设计

## 1.  概述

### 1.1 目的

保存所有OBD 设备链接信息和OBD Server的运行时信息，供后端系统对OBD设备和OBD Server的快速检索。本系统选用Redis作为缓存。

### 1.2 适用范围

适用于公车管理系统开发人员。

## 2.  Redis简介

※ redis是一个key-value存储系统。和Memcached类似，它支持存储的value类型相对更多，包括string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和hash（哈希类型）。这些数据类型都支持push/pop、add/remove及取交集、并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis支持各种不同方式的排序。为了保证效率，数据都是缓存在内存中，redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了master-slave(主从)同步。

※ Redis 是一个高性能的key-value数据库。在部分场合可以对关系数据库起到很好的补充作用。它提供了Java，C/C++，C#，PHP，JavaScript，Perl，Object-C，Python，Ruby，Erlang等客户端，使用很方便。

### 2.1存储方式

缓存采用KEY/VALUE的存储方式。

### 2.2 数据结构

redis提供五种数据类型：[string](http://baike.baidu.com/view/1005388.htm)，hash，list，set及zset(sorted set)。

1. **Strings - 字符串**

Redis的字符串是字节序列。在Redis中字符串是二进制安全的，这意味着他们有一个已知的长度，是没有任何特殊字符终止决定的，所以可以存储任何东西，最大长度可达512兆。

redis 127.0.0.1:6379> SET name "yiibai"

OK

redis 127.0.0.1:6379> GET name

"zzz"

1. **Hashes - 哈希值**

Redis的哈希键值对的集合。 Redis的哈希值是字符串字段和字符串值之间的映射，所以它们被用来表示对象

redis 127.0.0.1:6379> HMSET user:1 username zzz password 111 points 200

OK

redis 127.0.0.1:6379> HGETALL user:1

1) "username"

2) " zzz "

3) "password"

4) "111"

5) "points"

6) "200"

1. **Lists - 列表**

Redis的列表是简单的字符串列表，排序插入顺序。可以添加元素到Redis列表的头部或尾部。

redis 127.0.0.1:6379> lpush tutoriallist redis

(integer) 1

redis 127.0.0.1:6379> lpush tutoriallist mongodb

(integer) 2

redis 127.0.0.1:6379> lpush tutoriallist rabitmq

(integer) 3

redis 127.0.0.1:6379> lrange tutoriallist 0 10

1) "rabitmq"

2) "mongodb"

3) "redis"

1. **Sets - 集合**

Redis集合是字符串的无序集合。在Redis中可以添加，删除和测试文件是否存在在O(1)的时间复杂度的成员。

redis 127.0.0.1:6379> sadd tutoriallist redis

(integer) 1

redis 127.0.0.1:6379> sadd tutoriallist mongodb

(integer) 1

redis 127.0.0.1:6379> sadd tutoriallist rabitmq

(integer) 1

redis 127.0.0.1:6379> sadd tutoriallist rabitmq

(integer) 0

redis 127.0.0.1:6379> smembers tutoriallist

1) "rabitmq"

2) "mongodb"

3) "redis"

1. **集合排序**

Redis的集合排序类似于Redis集合，字符串不重复的集合。不同的是，一个有序集合的每个成员关联分数，用于以便采取有序set命令，从最小的到最大的分数有关。虽然成员都是独一无二的，分数可能会重复。

redis 127.0.0.1:6379> zadd tutoriallist 0 redis

(integer) 1

redis 127.0.0.1:6379> zadd tutoriallist 0 mongodb

(integer) 1

redis 127.0.0.1:6379> zadd tutoriallist 0 rabitmq

(integer) 1

redis 127.0.0.1:6379> zadd tutoriallist 0 rabitmq

(integer) 0

redis 127.0.0.1:6379> ZRANGEBYSCORE tutoriallist 0 1000

1) "redis"

2) "mongodb"

3) "rabitmq"

## 3.  OBD数据结构

### 3.1服务程序运行时信息

记录通信服务器运行信息

1. 数据库KEY值

格式OBDSERVER\_[IP]:[PORT]

IP：服务程序地址

PORT：服务程序监听端口

例如

OBDSERVER\_127.0.0.1:8907

1. 数据库索引ID

对于服务程序为固定值1df20ae0-0a6e-4b8e-ae03-92e1eb5947f1

1. 数据库VALUE值

格式：Json格式的ObdCacheDataServer数据结构。

Class ObdCacheDataServer

{

String ipAddress;

int port;

int httpPort;

int capacity;

Date updatedDatetime;

int clientCount;

}

ipAddress：OBD Server服务器地址

port：OBD Server服务器端口

httpPort：本地监听的HTTP端口

capacity：OBD Server能容纳的最大客户端数量

updatedDatetime：OBD Server状态更新时间

clientCount：当前维护链接的OBD设备数量

### 3.2 OBD设备链接信息

记录OBD设备链接信息

1. 数据库KEY值

格式OBDSERVER\_[IP]:[PORT]

IP：服务程序地址

PORT：服务程序监听端口

例如

OBDSERVER\_127.0.0.1:8907

1. 数据库索引ID

OBD设备的OBD Number

1. 数据库VALUE值

OBD设备的Channel ID