下载安装redis

地址忘了。有空找找

解压redis-windows-master.zip，在downloads目录下解压需要的版本(如redis64-3.0.501.zip)

运行windows命令行(cmd),进入Redis64-3.0.501所在目录

执行启动命令 redis-server.exe redis.windows.conf

至此redis服务在windows上启动成功，如需关闭，可直接关闭启动命令的窗口

另外测试可重新启动一个命令行窗口，进入目录，执行以下命令

redis-cli.exe

set age 21

get age

查看结果

c/c++利用hiredis操作访问redis

下载所需版本的源码https://github.com/MSOpenTech/redis

解压，进入msvs目录，使用VS2013打开RedisServer.sln

修改hiredis和Win32\_Interop项目编译环境：

选择x64平台

项目属性->c/c++->代码生成->运行库Debug为MDd, Release为MD

注意如实际工程使用的是MT或MTd,以上可以不做修改

修改debug输出的文件名，默认和release相同，建议\_d区分

编译生成hiredis\_d.lib，hiredis.lib，Win32\_Interop\_d.lib，Win32\_Interop.lib，debug和release

新建实际使用的win32控制台项目

复制并包含deps\hiredis和src\Win32\_Interop下的所有头文件到项目包含目录，注意也包括文件夹本身，可直接复制整个项目提出其他文件，之保留.h头文件即可

复制编译好的lib到指定目录

修改编译环境：

项目属性->c/c++->附加包含目录 $(SolutionDir)Include $(SolutionDir)Include\src $(SolutionDir)Include\deps\hiredis

项目属性->c/c++->预编译头 不使用预编译头

连接器->常规->附加库目录 添加lib所在目录

连接器->输入->附加依赖项 添加对应 hiredis\_d.lib;Win32\_Interop\_d.lib

连接器->输入->忽略特定默认库 libcmt.lib

至此可在vs2013中C++操作redis

测试代码如下: 貌似binary的有些错误，但能正常的set和get了

#include "stdafx.h"

#include <stdlib.h>

#include <string.h>

#include <hiredis.h>

#include <WinSock2.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

unsigned int j;

redisContext \*c;

redisReply \*reply;

struct timeval timeout = { 1, 500000 }; // 1.5 seconds

c = redisConnectWithTimeout((char\*)"127.0.0.1", 6379, timeout);

if (c->err) {

printf("Connection error: %s\n", c->errstr);

exit(1);

}

/\* PING server \*/

reply = (redisReply \*)redisCommand(c, "PING");

printf("PING: %s\n", reply->str);

freeReplyObject(reply);

/\* Set a key \*/

reply = (redisReply \*)redisCommand(c, "SET %s %s", "foo", "hello world");

printf("SET: %s\n", reply->str);

freeReplyObject(reply);

/\* Try a GET and two INCR \*/

reply = (redisReply \*)redisCommand(c, "GET foo");

printf("GET foo: %s\n", reply->str);

freeReplyObject(reply);

// /\* Set a key using binary safe API \*/

// reply = (redisReply \*)redisCommand(c, "SET %b %b", "bar", 3, "hello", 5);

// printf("SET (binary API): %s\n", reply->str);

// freeReplyObject(reply);

reply = (redisReply \*)redisCommand(c, "INCR counter");

printf("INCR counter: %lld\n", reply->integer);

freeReplyObject(reply);

/\* again ... \*/

reply = (redisReply \*)redisCommand(c, "INCR counter");

printf("INCR counter: %lld\n", reply->integer);

freeReplyObject(reply);

/\* Create a list of numbers, from 0 to 9 \*/

reply = (redisReply \*)redisCommand(c, "DEL mylist");

freeReplyObject(reply);

for (j = 0; j < 10; j++) {

char buf[64];

sprintf\_s(buf, 64, "%d", j);

reply = (redisReply \*)redisCommand(c, "LPUSH mylist element-%s", buf);

freeReplyObject(reply);

}

/\* Let's check what we have inside the list \*/

reply = (redisReply \*)redisCommand(c, "LRANGE mylist 0 -1");

if (reply->type == REDIS\_REPLY\_ARRAY) {

for (j = 0; j < reply->elements; j++) {

printf("%u) %s\n", j, reply->element[j]->str);

getchar();

}

}

freeReplyObject(reply);

return 0;

}