压缩态内存数据库实时算法设计与实现

用户使用手册

V1.0

小组名称：Never give up
小组口号：Make the change
指导教师：赵振刚老师
文档撰写人：蓝鸿翔
文档撰写时间：2014 年 6 月 5 日

团队分工记录表

项目名称	学号	姓名	分工
压缩态内存数据库实时算法设计与实现	SG13225025	蓝鸿翔	用户使用手册
	SA13226282	刘勇	
	SG13225022	范亚林	

目录

1. 系统概述

1.1 产品特点

随着互联网技术的发展、 硬件的不断更新、 企业及政府信息化的不断深入, 应用的复杂性要求越来越高, 推动着数据存储技术向着海量数据、 分析数据、 智能数据的方向发展, 以便为数据仓库、 在线分析提供高效实时的技术支持。基于行存储的数据库技术面临新的问题,已经出现了技术瓶颈。近些年来, 一种新的数据存储理念, 即基于列存储的关系型数据库(简称列数据库, 下同)应运而生。列数据库能够快速发展, 主要原因是其复杂查询效率高, 读磁盘少,存储空间少, 以及由此带来的技术、 管理和应用优势。对列数据库技术的基本现状、 关键支撑技术以及应用优势进行了介绍和分析。

本系统采用列压缩技术，对数据分别进行游程压缩，比特压缩，字典压缩，空值悬挂、原始数据、从而有效地减小数据的存储空间，同时设计在压缩态数据集合上操作存取操作，使得在不解压的情况下对数据进行存取操作。

1.2 本产品应用领域和对象

本产品可适用于：

1、适用于海量数据的大型服务器

2、适用于存储空间昂贵的嵌入式平台

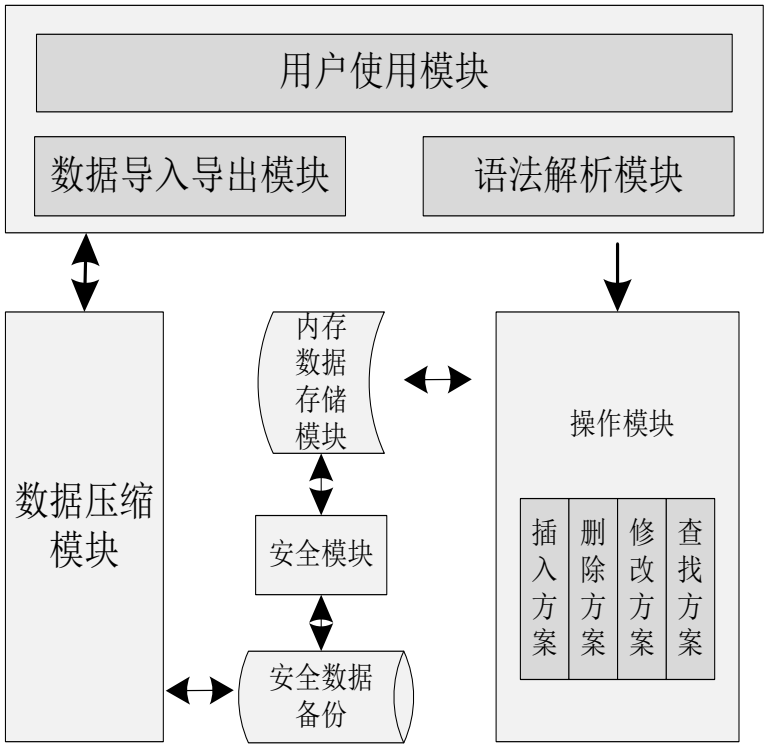
3、适用于对删改操作要求不高的用户

4、适用于查询密集型用户

2. 系统综述

2.1 系统结构

本系统主要由几个模块组成：数据导入模块，数据属性统计模块，数据压缩模块，数据操作模块，压缩数据导出模块：



2.2 系统使用/开发环境

计算机型号：联想Z475

CPU：AMD Llano APU A4-3300M ，1.4GHZ

内存：4GB DDR3 1333MHz

操作系统：linux ubuntu 12.04

2.3 系统部署与使用

2.3.1 源代码编译

（1）进入到工程目录：

./config 配置当前环境

```
lanhxg@Lenovo:mysqlite_v20140528$ ./config.sh
export LD_LIBRARY_PATH=/usr/src/work/TeamWork/mysqlite_v20140528/lib
lanhxg@Lenovo:mysqlite_v20140528$ export LD_LIBRARY_PATH=/usr/src/work/TeamWork/mysqlite_v20140528/lib
```

（2）利用工程目录下的Makefile直接生产工程Demo

make clean;make

```
lanhxg@Lenovo:mysqlite_v20140528$ make clean;make
rm -f apptable.o bvector.o csv.o dbsystem.o diction.o main.o nullmark.o original.o
.time *.mem ;clear
```

可以看到 Demo程序的名字是mysqlite

```
lanhxg@Lenovo:mysqlite_v20140528$ ls
apptable.c  bvector.c  config.sh  csv.o     dbsystem.o  diction.h  log     Makefile   nullmark.c  original.c  rbtree.c  runlength.c
apptable.h  bvector.h  csv.c      dbsystem.c  debug.h     diction.o  main.c  mhistory.d  nullmark.h  original.h  rbtree.h  runlength.h
apptable.o  bvector.o  csv.h      dbsystem.h  diction.c   lib        main.o  mysqlite    nullmark.o  original.o  rbtree.o  runlength.o
lanhxg@Lenovo:mysqlite_v20140528$
```

(3)执行Demo

./mysqlite

2.3.2 测试数据导入

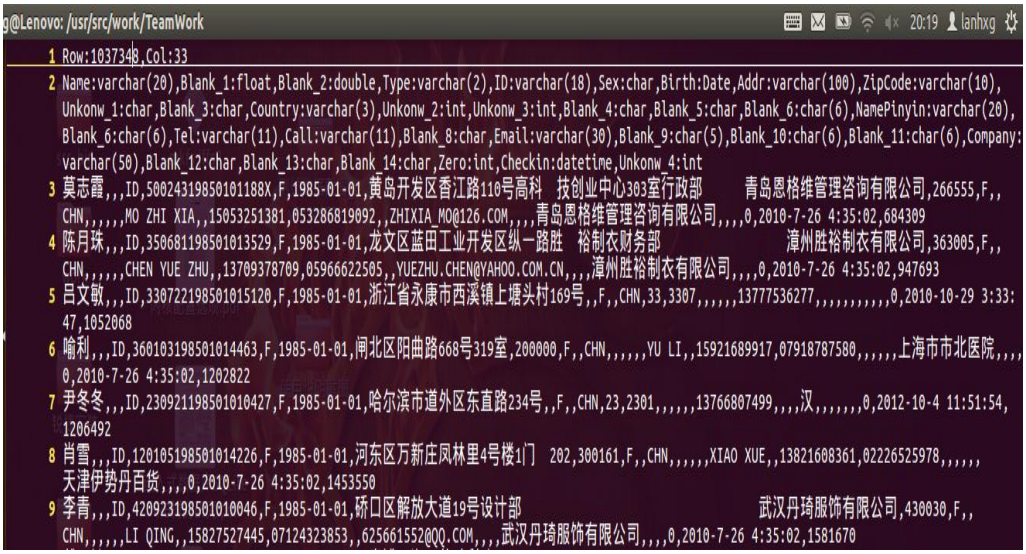
（1）对于导入的数据，采用csv格式文件，默认的命名方式是log

（2）文件格式要求：

首行：必须说明该csv文件包含的列数和行数（不包括最前面两个属性行）

次行：必须说明该csv文件每列的数据类型，名字，数据的长度

其余：存放表格数据，以逗号分隔数据单元



3. 测试 Demo 配置使用

本章主要说明如何进行压缩测试、读取测试、数据追加测试：压缩态数据读取测试，追加数据部分读取测试。

3.1 测试配置文件

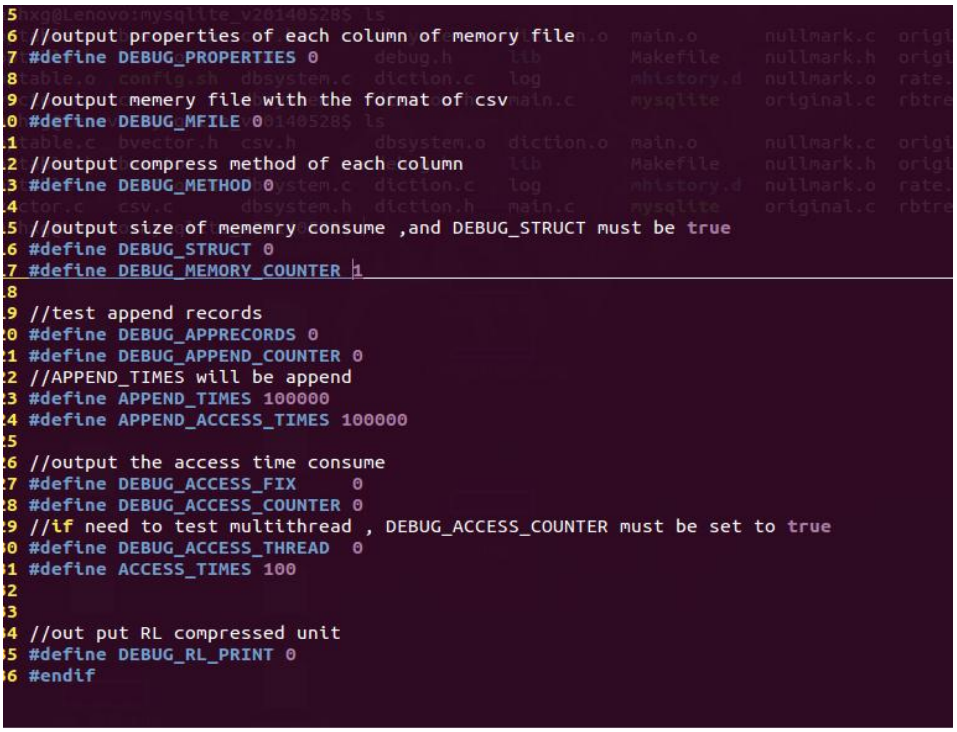
对于所有的模块测试，都在工程目录下的 debug.h 文件中通过配置选项来确定：

```
//output properties of each column of memory file
#define DEBUG_PROPERTIES 0

//output memery file with the format of csv
#define DEBUG_MFILE 0

//output compress method of each column
#define DEBUG_METHOD 0

#define DEBUG_STRUCT 0
//output size of mememry consume ,and DEBUG_STRUCT must be true
#define DEBUG_MEMORY_COUNTER 1

//test append records
#define DEBUG_APPRECORDS 1000
#define APPEND_TIMES 100000
#define APPEND_ACCESS_TIMES 100000
#define DEBUG_APPEND_COUNTER 1

//output the access time consume
#define DEBUG_ACCESS_FIX    1
#define DEBUG_ACCESS_COUNTER 0
//if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
#define DEBUG_ACCESS_THREAD  0
#define ACCESS_TIMES 100

//out put RL compressed unit
#define DEBUG_RL_PRINT 0
```

3.1.1 压缩比 Demo 配置测试

1. 配置选项图

只需要把选项配置成对应的值即可:



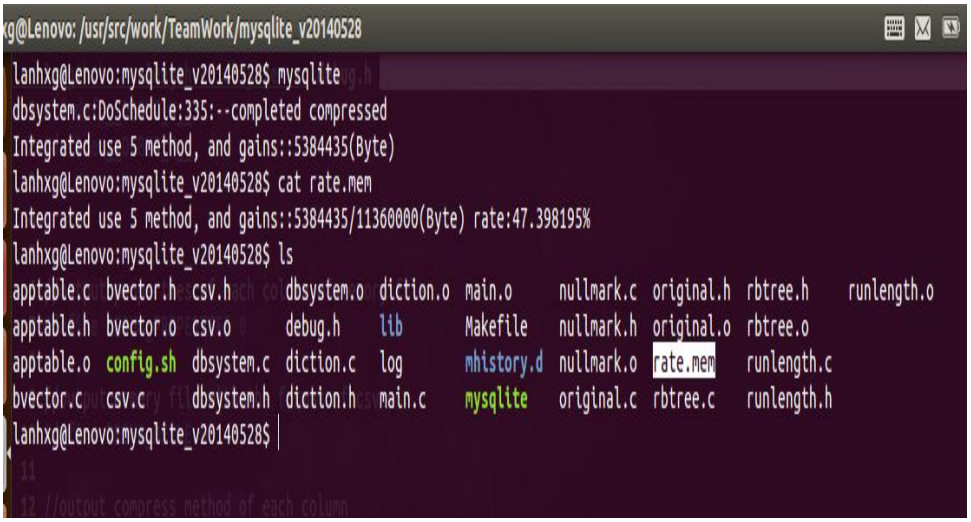
```
//output properties of each column of memory file
#define DEBUG_PROPERTIES 0
//output memery file with the format of csv
#define DEBUG_MFILE 0
//output compress method of each column
#define DEBUG_METHOD 0
//output size of mememry consume ,and DEBUG_STRUCT must be true
#define DEBUG_STRUCT 0
#define DEBUG_MEMORY_COUNTER 1

//test append records
#define DEBUG_APPRECORDS 0
#define DEBUG_APPEND_COUNTER 0
//APPEND_TIMES will be append
#define APPEND_TIMES 100000
#define APPEND_ACCESS_TIMES 100000

//output the access time consume
#define DEBUG_ACCESS_FIX     0
#define DEBUG_ACCESS_COUNTER 0
//if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
#define DEBUG_ACCESS_THREAD  0
#define ACCESS_TIMES 100


//out put RL compressed unit
#define DEBUG_RL_PRINT 0
#endif
```

2. 重新生成工程文件

```
Make clean;make
```

3. 执行并查看输出

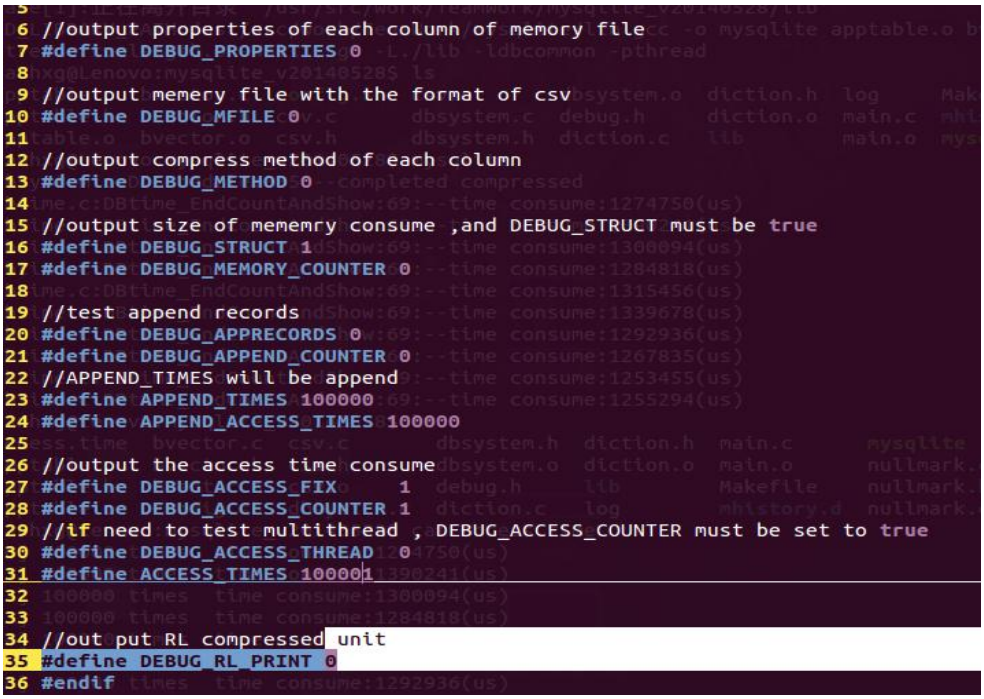
```
./mysqlite

ls -lrht
```



3.1.2 压缩态查询数据效率测试

1. 配置选项图

只需要把选项配置成对应的值即可：



2. 重新生成工程文件

Make clean;make

3. 执行并查看输出

./mysqlite

ls –lrht

```
LD_LIBRARY_PATH=/usr/src/work/TeamWork/mysqlite/lib gcc -o mysqlite apptable.o bvector.o csv.o dbsystem.o diction.o main.o nullmark.o o
btree.o runlength.o -Wall -g -L./lib -ldbcommon -pthread
lanhxg@Lenovo:mysqlite_v20140528$ ls
apptable.c  bvector.c  config.sh  csv.o     dbsystem.o  diction.h  log     Makefile    nullmark.c  original.c  rbtree.c  runlength.c
apptable.h  bvector.h  csv.c      dbsystem.c  debug.h     diction.o  main.c  mhistory.d  nullmark.h  original.h  rbtree.h  runlength.h
apptable.o  bvector.o  csv.h      dbsystem.h  diction.c   lib        main.o  mysqlite    nullmark.o  original.o  rbtree.o  runlength.o
lanhxg@Lenovo:mysqlite_v20140528$ mysqlite
dbsystem.c:DoSchedule:335:--completed compressed
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1274750(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1390241(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1300094(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1284818(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1315456(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1339678(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1292936(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1267835(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1253455(us)
dbtime.c:DBtime_EndCountAndShow:69:--time consume:1255294(us)
lanhxg@Lenovo:mysqlite_v20140528$ ls
access.time  bvector.c  csv.c       dbsystem.h  diction.h  main.c      mysqlite    original.c  rbtree.h     runlength.o
apptable.c   bvector.h  csv.h       dbsystem.o  diction.o  main.o      nullmark.c  original.h  rbtree.o
apptable.h   bvector.o  csv.o       debug.h     lib        Makefile    nullmark.h  original.o  runlength.c
apptable.o   config.sh  dbsystem.c  diction.c   log        mhistory.d  nullmark.o  rbtree.c    runlength.h
lanhxg@Lenovo:mysqlite_v20140528$ cat access.time
[0] 100000 times  time consume:1274750(us)
[1] 100000 times  time consume:1390241(us)
[2] 100000 times  time consume:1300094(us)
[3] 100000 times  time consume:1284818(us)
[4] 100000 times  time consume:1315456(us)
[5] 100000 times  time consume:1339678(us)
[6] 100000 times  time consume:1292936(us)
[7] 100000 times  time consume:1267835(us)
[8] 100000 times  time consume:1253455(us)
[9] 100000 times  time consume:1255294(us)
```

3.1.3 原始态数据追加效率测试

1. 配置选项图

只需要把选项配置成对应的值即可：

```
 4
 5
 6 //output properties of each column of memory file
 7 #define DEBUG_PROPERTIES 0
 8
 9 //output memery file with the format of csv
10 #define DEBUG_MFILE 0
11
12 //output compress method of each column
13 #define DEBUG_METHOD 0
14
15 //output size of mememry consume ,and DEBUG_STRUCT must be true
16 #define DEBUG_STRUCT 1
17 #define DEBUG_MEMORY_COUNTER 0
18
19 //test append records
20 #define DEBUG_APPRECORDS 0
21 #define DEBUG_APPEND_COUNTER 1
22 //APPEND_TIMES will be append
23 #define APPEND_TIMES 100000
24 #define APPEND_ACCESS_TIMES 100000
25
26 //output the access time consume
27 #define DEBUG_ACCESS_FIX      0
28 #define DEBUG_ACCESS_COUNTER 0
29 //if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
30 #define DEBUG_ACCESS_THREAD   0
31 #define ACCESS_TIMES 100001
32
33
34 //out put RL compressed unit
35 #define DEBUG_RL_PRINT 0
```

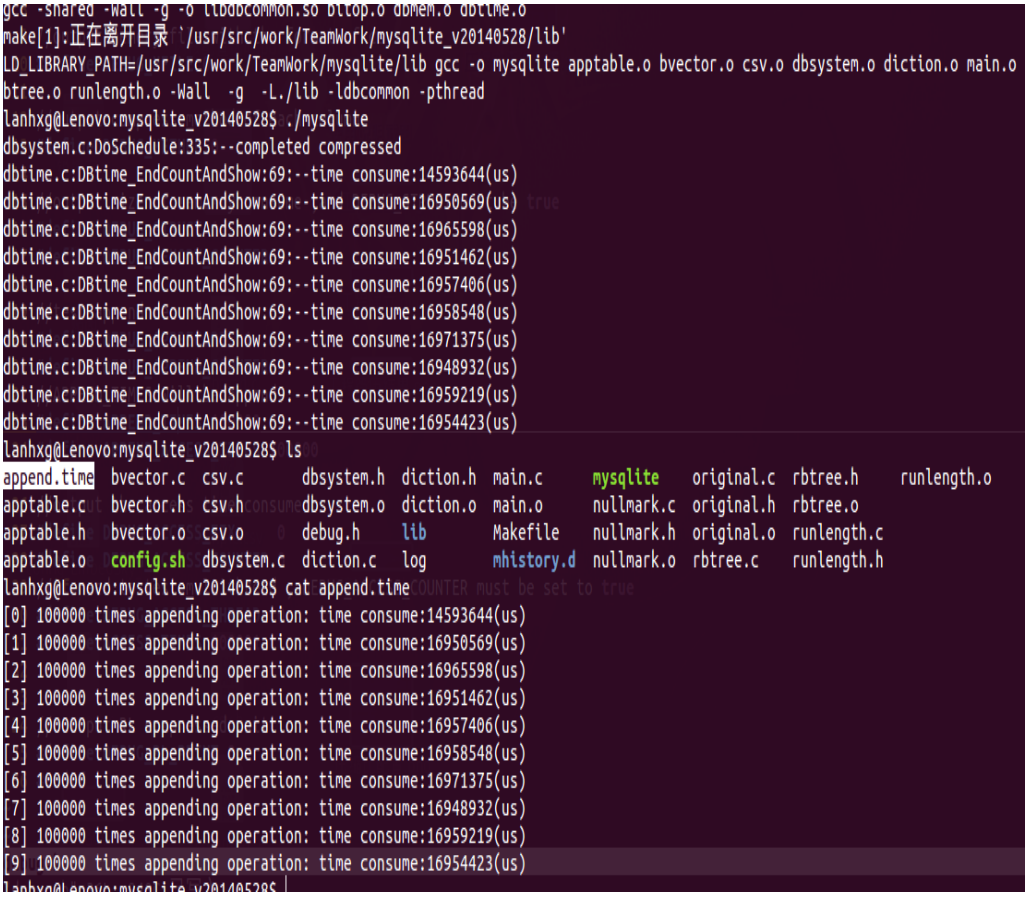
2. 重新生成工程文件

Make clean;make

3. 执行并查看输出

./mysqlite

ls –lrht



3.1.4 追加数据查询效率测试

1. 配置选项图

只需要把选项配置成对应的值即可：

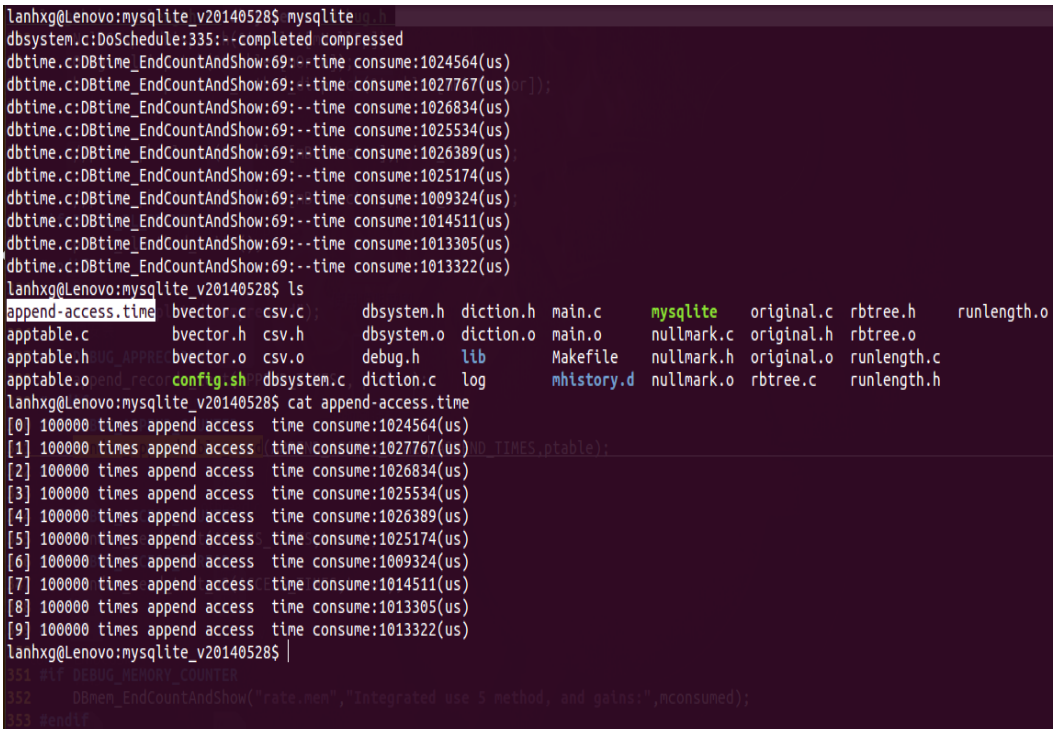
```
 4
 5
 6 //output properties of each column of memory file
 7 #define DEBUG_PROPERTIES 0
 8
 9 //output memery file with the format of csv
10 #define DEBUG_MFILE 0
11
12 //output compress method of each column
13 #define DEBUG_METHOD 0
14
15 //output size of mememry consume ,and DEBUG_STRUCT must be true
16 #define DEBUG_STRUCT 1
17 #define DEBUG_MEMORY_COUNTER 0
18
19 //test append records
20 #define DEBUG_APPRECORDS 0
21 #define DEBUG_APPEND_COUNTER 1
22 //APPEND_TIMES will be append
23 #define APPEND_TIMES 100000
24 #define APPEND_ACCESS_TIMES 100000
25
26 //output the access time consume
27 #define DEBUG_ACCESS_FIX      0
28 #define DEBUG_ACCESS_COUNTER 0
29 //if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
30 #define DEBUG_ACCESS_THREAD   0
31 #define ACCESS_TIMES 100001
32
33
34 //out put RL compressed unit
35 #define DEBUG_RL_PRINT 0
```

2. 重新生成工程文件

```
Make clean;make
```

3. 执行并查看输出

```
./mysqlite

ls –lrht
```



3.1.5 数据导出测试

1. 配置选项图

只需要把选项配置成对应的值即可：

```
//output properties of each column of memory file
#define DEBUG_PROPERTIES 0

//output memery file with the format of csv
#define DEBUG_MFILE 1

//output compress method of each column
#define DEBUG_METHOD 0

//output size of mememry consume ,and DEBUG_STRUCT must be true
#define DEBUG_STRUCT 1
#define DEBUG_MEMORY_COUNTER 0

//test append records
#define DEBUG_APPRECORDS 0
#define DEBUG_APPEND_COUNTER 0
//APPEND_TIMES will be append
#define APPEND_TIMES 100000
#define APPEND_ACCESS_TIMES 100000

//output the access time consume
#define DEBUG_ACCESS_FIX      0
#define DEBUG_ACCESS_COUNTER 0
//if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
#define DEBUG_ACCESS_THREAD  0
#define ACCESS_TIMES 100000

//out put RL compressed unit
#define DEBUG_RL_PRINT 0
#endif
```

2. 重新生成工程文件

Make clean;make

3. 执行并查看输出

./mysqlite

ls -lrht

```
lanhxg@Lenovo:mysqlite_v20140528$ mysqlite >outlog
lanhxg@Lenovo:mysqlite_v20140528$ head -n 10 outlog
Name,Blank_1,Blank_2,Type,ID,Sex,Birth,Addr,ZipCode,Unkonw_1,Blank_3,Country,Unkonw_2,Unkonw_3,Blank_4,Blank_5,Blank_6,NamePinyin,Blank_6,Tel,Call
,Blank_8,Email,Blank_9,Blank_10,Blank_11,Company,Blank_12,Blank_13,Blank_14,Zero,Checkin,Unkonw_4
莫志霞,,,ID,500243198501011888X,F,1985-1-1,黄岛开发区香江路110号高科技创业中心303室行政部青岛恩格维管理咨询有限公司,266555,F,,CHN,,,,,MOZHIXIA,,15
053251381,053286819092,,ZHIXIA_MO@126.COM,,,,青岛恩格维管理咨询有限公司,,,,0,2010-7-8 0:0:0,684309
陈月珠,,,ID,350681198501013529,F,1985-1-1,龙文区蓝田工业开发区纵一路胜裕制衣财务部漳州胜裕制衣有限公司,363005,F,,CHN,,,,,CHENYUEZHU,,13709378709,
05966622505,,YUEZHU.CHEN@YAHOO.COM.CN,,,,漳州胜裕制衣有限公司,,,,0,2010-7-8 0:0:0,947693
吕文敏,,,ID,330722198501015120,F,1985-1-1,浙江省永康市西溪镇上塘头村169号,,F,,CHN,33,3307,,,,,13777536277,,,,,,,,,,0,2010-10-37 0:0:0,1052068
喻利,,,ID,360103198501014463,F,1985-1-1,闸北区阳曲路668号319室,200000,F,,CHN,,,,,YULI,,15921689917,07918787580,,,,,,上海市市北医院,,,,0,2010-7-8
0:0:0,1202822
尹冬冬,,,ID,230921198501010427,F,1985-1-1,哈尔滨市道外区东直路234号,,F,,CHN,23,2301,,,,,,13766807499,,,,汉,,,,,,,0,2012-10-155 0:0:0,1206492
肖雪,,,ID,120105198501014226,F,1985-1-1,河东区万新庄凤林里4号楼1门202,300161,F,,CHN,,,,,,XIAOXUE,,13821608361,02226525978,,,,,,天津伊势丹百货,,,,0
,2010-7-8 0:0:0,1453550
李青,,,ID,420923198501010046,F,1985-1-1,硚口区解放大道19号设计部武汉丹琦服饰有限公司,430030,F,,CHN,,,,,,LIQING,,15827527445,07124323853,,625661552
@QQ.COM,,,,武汉丹琦服饰有限公司,,,,0,2010-7-8 0:0:0,1581670
戴元敏,,,ID,310229198501011422,F,1985-1-1,青浦区海天苑胜利路295弄15号501,201700,F,,CHN,,,,,,DAIYUANMIN,,13564100263,02169201875,,181613301@QQ.COM,
,,,上海同策房产咨询股份有限公司,,,,0,2010-7-8 0:0:0,1836496
汪景丹,,,ID,210881198501012542,F,1985-1-1,开发区保税区洞庭路1号石化大厦616大连赤兔马贸易有限公司营销,116600,F,,CHN,,,,,,WANGJINGDAN,,13889641304,0
41187540240,,WANGJINGDAN47@HOTMAILCOM,,,,大连赤兔马贸易有限公司,,,,0,2010-7-8 0:0:0,1839492
lanhxg@Lenovo:mysqlite_v20140528$
```

3.1.6 压缩态多线程查询效率测试

1. 配置选项图

只需要把选项配置成对应的值即可：

```
//output properties of each column of memory file
#define DEBUG_PROPERTIES 0
//output memery file with the format of csv
#define DEBUG_MFILE 0
//output compress method of each column
#define DEBUG_METHOD 0
//output size of mememry consume ,and DEBUG_STRUCT must be true
#define DEBUG_STRUCT 1
#define DEBUG_MEMORY_COUNTER 0
//test append records
#define DEBUG_APPRECORDS 0
#define DEBUG_APPEND_COUNTER 0
//APPEND_TIMES will be append
#define APPEND_TIMES 100000
#define APPEND_ACCESS_TIMES 100000
//output the access time consume
#define DEBUG_ACCESS_FIX    1
#define DEBUG_ACCESS_COUNTER 1
//if need to test multithread , DEBUG_ACCESS_COUNTER must be set to true
#define DEBUG_ACCESS_THREAD  1
#define ACCESS_TIMES 100001
```

2. 重新生成工程文件

Make clean;make

3. 执行并查看输出

./mysqlite

ls –lrht

```
anhxg@Lenovo:mysqlite_v20140528$ ls
ccess.time  bvector.c  csv.c      dbsystem.h  diction.h  main.c     mysqlite    original.c  rbtree.h     runlength.o
pptable.c   bvector.h  csv.h      dbsystem.o  diction.o  main.o     nullmark.c  original.h  rbtree.o     thread.time
pptable.h   bvector.o  csv.o      debug.h     lib        Makefile   nullmark.h  original.o  runlength.c
pptable.o   config.sh  dbsystem.c diction.c   log        mhistory.d nullmark.o  rbtree.c    runlength.h
anhxg@Lenovo:mysqlite_v20140528$ cat thread.time
0] 100001 times time consume:783012(us)
1] 100001 times time consume:777468(us)
2] 100001 times time consume:779336(us)
3] 100001 times time consume:805388(us)
4] 100001 times time consume:780249(us)
5] 100001 times time consume:779918(us)
6] 100001 times time consume:779078(us)
7] 100001 times time consume:777307(us)
8] 100001 times time consume:780195(us)
9] 100001 times time consume:778230(us)
```