

gStore2.0

Building:

Install gStore2.0, just three commands.

System Requirement: 64bit linux with g++, gcc, make installed

① uncompress the gStore.tar.gz and get into the gStore folder

```
tar -zxvf gStore.tar.gz
```

② run

```
make lib_antlr
```

to compile and link the library(static) we need

③ run

```
make
```

to compile the gStore code and build two executable “gload” and “gquery” .

gStore2.0 must be built using GNU make and a reasonable c++ compiler.

Using gStore2.0:

gStore2.0 currently includes two executables.

1. The first (gStore2.0) is used to build a new database from a 3-triples input:

```
./gload db_name rdfdata.n3 (see Figure 1)
```

Make sure that: (see Figure 2)

① Each line in the input RDF data file end up with a dot.

② Each line only contains a triple, whose tuples(subject, predicate and object) separated by tab.

For example: (yago_10000 and query_10000 are both in dir gStore/example/, test with it)

yago_10000 is made up with the first 10,000 lines of yago dataset.

Figure 1

```
[root@master Gstore]# ./gload db_yago10000 ./example/yago_10000
2014年11月13日 星期四 21时42分17秒 -0.437969 seconds
gload...
argc: 3 DB_store:db_yago10000 RDF_data: ./example/yago_10000
begin encode RDF from : ./example/yago_10000 ...
```

Figure 2

```
<!!!> y:created <Louden_Up_Now>.
<!!!> y:created <!!!_(album)>.
<!!!> y:created <Take_Ecstasy_with_Me%2FGet_Up>.
<!!!> y:created <Myth_Takes>.
<!!!> y:establishedOnDate <1996-%23%23-%23%23>.
<!!!> y:hasWebsite <http://www.chkchkchk.net>.
<!!!> y:isOfGenre <wordnet_revival_101047338>.
<!!!> rdfs:label "\u0021\u0021\u0021".
```

2. After loading the database can be queried with the second executable: gquery

`./gquery db_name` (see Figure4)

The program shows a command prompt: inputting a file name that can be used to interpret a file with single query(see Figure 3). When the program finish answering the query(see Figure 5), it shows the command prompt again.

gStore2.0 only support simple "select" queries.

Figure 3

```
[root@master Gstore]# cat example/query_10000
select ?x1 ?x2 where
{
?x1      y:created      ?x2.
?x1      y:created      <Louden_Up_Now>.
?x2      y:hasSuccessor <Louden_Up_Now>.
?x2      rdf:type       <wordnet_album_106591815>.
}
```

Figure 4

```
[root@master Gstore]# ./gquery db_yago10000
gquery...
argc: 2 DB_store:db_yago10000
loadTree...
LRUCache initial...
LRUCache initial finish
finish loadCache
finish loadEntityID2FileLineMap
open KVstore
finish load
finish loading
please input query file path:
example/query_10000|
```

Figure5

```
Total time used: 1ms.
final result is :
There has answer: 1
?x1      ?x2
<!!!>    <!!!_(album)>

please input query file path:
|
```