TripleSlogS: A Java-based Tool for Profiling Logs from Enterprise RDF Stores

Ghislain A. Atemezing†,

†Mondeca, R&D
35, Boulevard de Strasbourg, 75010, Paris, France
ghislain.atemezing@mondeca.com / @gatemezing

Mining logs from Enterprise RDF Stores can be very useful to quickly grasp the nature of the SPARQL queries and generate various statistics.

1. You want to monitor SPARQL queries logs generated from your endpoint
2. You need a simple tool to quickly understand users' queries
3. You use CLI for your daily semantic tasks activities

TripleSlogS [2] is the tool you need: easy to use, output different types of statistics, open source and extensible to other Graph Databases Logs

Dude, we've collected a quite number of logs from Virtuoso and GraphDB enpoints in my company?

Great man! What are you going to do with such logs?

Well, my manager asked me to make some statistics? But I can't find any open source tool out there? Any idea?

Interesting! I might have a solution fo you? It's a JAVA tool to do exactly what you need?

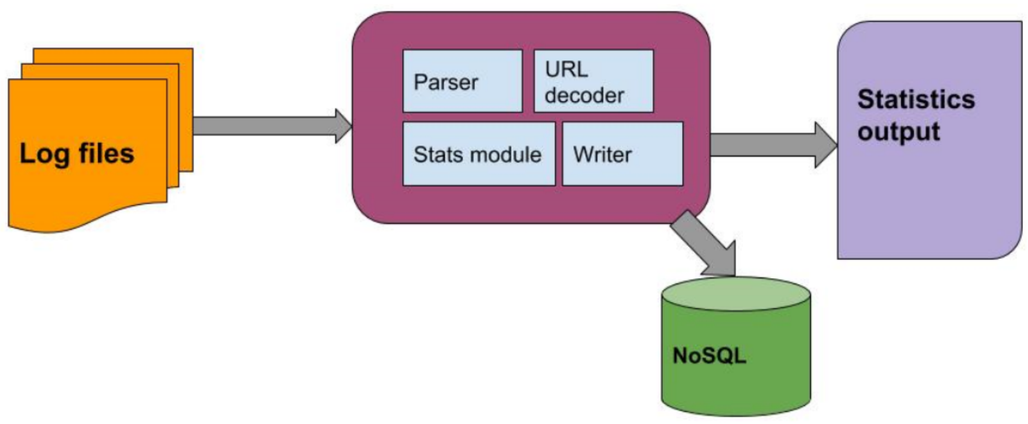
Dude, please send me the link. I'll be happy to give it trial. TIA.

Here you go [1]. Please send me any feedback. Hope it helps!

Sure. Thanks Dude! You saved my day!! ☺

TripleSlogS Architecture

- The **parser** handles log files detects a valid SPARQL query and the status of HTTP requests.
- The **URL decoder/encoder** module is used to perform some validation checks
- The **Stats module** is used for gathering the different computations covered by the tool, and aggregation functions available.
- The **Writer** produces various outputs and connects to NoSQL database.



SCAN ME

TripleSlogS Functionalities

- Current implementation for Virtuoso [3]and GraphDB [4].
- Stats on total number of all queries
- Number of distinct queries in format "{ N / M, p% }", where N is the number of distinct queries, M is the number of all queries and p% the percentage of distinct queries in the file.
- Number of Basic Graph Pattern (BGP) in all queries.
- Number and percentage of CONSTRUCT,ASK,DESCRIBE,SELECT queries
- Number of the main SPARQL constructions (UNION, DISTINCT, ORDER BY, REGEX, LIMIT, OFFSET, OPTIONAL, etc).
- Provide with the time and error code of the HTTP queries (4X,2X, 3X, etc.)

```
virtuoso_http01072016.log:  date = 2016/07/01, statistics = { total =  906, countBGP =  376, distincts = {  475/
906, 52.43% },  types = { CONSTRUCT = {    0/ 475,  0.00% },  ASK = {    0/ 475,  0.00% },  DESCRIBE = {   88/
475, 18.53% },  SELECT = {  387/ 475, 81.47% } },  commands = { OPTIONAL = {   0/766,  0.00% },  FILTER = {
28/766,  3.66% },  REGEX = {   5/766,  0.65% },  DISTINCT = {   2/766,  0.26% },  OFFSET = { 360/766, 47.00% },
LIMIT = { 368/766, 48.04% },  GROUP BY = {   0/766,  0.00% },  UNION = {   1/766,  0.13% },  ORDER BY = {
2/766,  0.26% } } }

******************************** STATISTICS ********************************

Duplicate queries:                       832 / 1458,  57.06%

Distinct queries - number of ASK:         17 /  626,   2.72%
Distinct queries - number of DESCRIBE:   146 /  626,  23.32%
Distinct queries - number of SELECT:     460 /  626,  73.48%
Distinct queries - number of CONSTRUCT:    0 /  626,   0.00%

********************************* TOTAL **********************************

     7 FILES                  with errors:  6
                  with distinct errors:  6
  1458 QUERIES                with errors:  encoded = 0, unparsed = {   72 / 1458,  4.94% }
                  with distinct errors:  encoded = 0, unparsed = {   53 /  626,  8.47% }

Total time: 00 m 09.954 s
```

1. https://labs.mondeca.com/tripleslogs
2. https://github.com/MondecaLabs/tripleSlogS
3. http://vos.openlinksw.com/owiki/wiki/VOS
4. http://graphdb.ontotext.com/