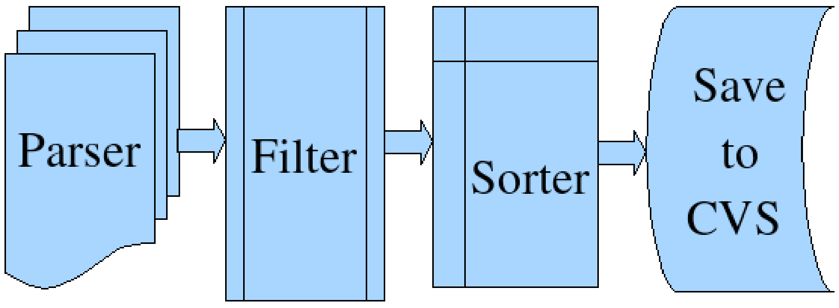
Table of Contents

Carmot Log Parser

How it works?

The main Java class is **com.yahoo.content.assembly.simulator.utility.LogParser**. Every log file it processed will go through following stage,

1. **Parser**: Parsing log message into a *HashMap*, specifying which files to parse by **-i** option.
2. **Filter**: Filtering by inspecting *HashMap*, by specifying **-F** option.
3. **Sorter**: Sorting by *HashMap*, by specifying **-S** option.
4. **Writer**: Dumping *HashMap* to a CSV file. You can specify which columns to save by specifying **-C** option.



Input files by specifying -i option

By default, the input file was set to **/tmp/cassandra/logs/system.log**

You can specify multiple input files, like

```
-i /tmp/cassandra/logs/system.log /tmp/cassandra/logs/system.log.2009-08-31-14
```

And, you can specify with wildcard,

```
-i /tmp/cassandra/logs/system.log*
```

💡 The input file can be the CSV file, which the output from this utlity. This is can be very useful, if you want to narrow down your investigation.

```
-i /tmp/cassandra/output-20090902-1157.csv
```

Parser

For example, a message like below, it will be parsed into

```
node1 TRACE [Timer-1] 2009-09-01 16:00:00,920 Gossiper.java (line 78) Performing status check ...
----- -----  -------  ----------------------- -------------       --  ---------------------------
node  level  thread   time                    file              line  body
```

It will be parsed into a *HashMap*, like

node	node1
level	TRACE
thread	Timer-1
time	2009-09-01 16:00:00,920
file	Gosiper.java
line	78

body	Performing status check ...

If the *body* contains pattern like **key:[value]**, the key/value pair will also be stored into *HashMap*.

If you want your message to be parsed by Log Parser, please use the pattern to log your message.

For example,

```
node1  INFO [MESSAGE-DESERIALIZER-POOL:1] 2009-08-26 10:37:36,001 SinkController.java (line 45) S
```

You will get a *HashMap* like,

node	node1
level	INFO
thread	MESSAGE-DESERIALIZER-POOL:1
time	2009-08-26 10:37:36,001
file	SinkController.java
line	45
STATE	BYPASS
FROM	10.72.180.62:7009
ID	2
VERB	ECHO_MESSAGE_VERB
MD5	42acffd34dd3bd50bd35c896c653c1a4

Filter by specifying -F option

If you don't include any **-F**, then every message will be matched, thus output CSV file will contains every message.

Log Parser provided five way to match the value in *HashMap*

1. **=**: check whether string is equal
2. **!=**: check whether string is not equal
3. **>**: check whether string is greater than
4. **<**: check whether string is less than
5. **>=**: check whether string is greater or equal than
6. **<=**: check whether string is less or equal than
7. **=~**: check whether string is match with Regex expression

With option **-F "node=node1&&body=~^Sending"**, it will be matching message like below

```
node1 TRACE [GMFD:1] 2009-09-01 17:00:00,926 Gossiper.java (line 996) Sending a GossipDigestAckMe
```

But, it won't match

```
node3 TRACE [GMFD:1] 2009-09-01 17:00:00,926 Gossiper.java (line 996) Sending a GossipDigestAckMe
```

You can specify option like **-F "node=node1&&body=~^Sending" "node=node3&&body=~^Gossip"**, which actually works like *OR*ing two or more filter. Thus, it will match with

```
node1 TRACE [GMFD:1] 2009-09-01 17:00:00,926 Gossiper.java (line 996) Sending a GossipDigestAckMe
node3 TRACE [Timer-1] 2009-09-01 17:00:01,557 Gossiper.java (line 343) Gossip Digests are : 192.1
```

Sorter by specifying -S option

You can specify which key in *HashMap* to sort with, like below, it will sort with **time** first, then with **node**

```
-S time node
```

By default, the sorting sequence is in **ascending**. If you want to sort with time descedningly, you can specify it like **time,D**, like

```
-S time,D node
```

Writer by specifying -C option

To specify which keys to be stored in CSV file. By default, it will include **node**, **time**, and **body**

```
-C node thread time body
```

Other command line option

Output filename by specifying -o option

By default, the output filename is set to **/tmp/cassandra/output-{CurrentTimeStamp}.csv**, like **/tmp/cassandra/output-20090902-1157.csv**. You can set it by

```
-o /tmp/output.csv
```

Delimiter for CVS file by specifying -d option

By default, delimiter is set to **;**, and it can set by

```
-d :
```

Too much to enter?

There're too much to enter in the command line, right? You can put all the options into a text file, then enter the fullpath for text file as the **first** argument.

For example, you can edit **/tmp/args.txt** to have following line,

```
-F "node=node1&&body=~^Sending" "node=node3&&body=~^Gossip" -C XXX node time nnnn body -S time no
```

Then, enter command line like

```
/tmp/args.txt -i /tmp/cassandra/logs/system.log*
```

Then, it is equivalent to

```
-F "node=node1&&body=~^Sending" "node=node3&&body=~^Gossip" -C XXX node time nnnn body -S time no
```

Sample Run

For example, if you like to check the message flow in the **MessageSink**, you can enter command line options like

```
-F body=~^SinkLog -S ID time node -C time node body STATE FROM ID VERB MD5
```

You will get something like,

```
#time;node;STATE;FROM;ID;VERB;MD5

2009-09-03 14:32:23,130;node1;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;992999960095711242l6
2009-09-03 14:32:23,131;node2;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:32:23,136;node3;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:32:23,140;node4;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:34:31,652;node0;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:34:31,654;node3;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:34:31,656;node1;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:34:31,671;node2;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216
2009-09-03 14:34:31,692;node4;SINK-STATE;10.72.180.62:7009;1;SINK_STATE_VERB;99299996009571124216

2009-09-03 14:33:00,320;node0;BYPASS;10.72.180.62:7009;2;RELAY_MESSAGE_VERB;44236794594009998280 6
2009-09-03 14:33:00,456;node0;OUTGO;10.72.180.62:7000;2;RESPONSE;886273115371499827703129194381 92
2009-09-03 14:34:31,717;node0;BYPASS;10.72.180.62:7009;2;RELAY_MESSAGE_VERB;44236794594009998280 6
2009-09-03 14:34:31,729;node0;OUTGO;10.72.180.62:7000;2;RESPONSE;886273115371499827703129194381 92
2009-09-03 14:58:28,411;node0;BYPASS;10.72.180.62:7009;2;RELAY_MESSAGE_VERB;44236794594009998280 6
2009-09-03 14:58:28,556;node0;OUTGO;10.72.180.62:7000;2;RESPONSE;886273115371499827703129194381 92
```

-- ShangyYahoo -- 01 Sep 2009

Last revised by: shangy on 18 Sep 2009

ATTACHMENT LIST
%ATTACHMENTLIST{format="$fileName?"}%