MySQL backup/restore

Vu Nguyen

Purpose

- For backup/restore disaster failure
- Ensure no data lost through “raw data” backup
- Defined sla base on data size (e.g 75gb, 500gb, 1TB, 5TB etc..)

Backup Design

- # Table per database -> e.g meeting_graph1 has 11 tables
- Table(Rows) / # of Threads
  - meeting_graph1.users has 2 392 810 rows
  - 2 392 810 / #cpu (at this time 24)
  - Generate 24 csv labeled thread#__timestamp__meeting_graph1_users.csv
  - Reduce the export time by ? with number of cores (Amdahl’s law) (https://en.m.wikipedia.org/wiki/Amdahl's_law)

Restore Design

```
For i in databasename.mysqltables:
    for i in numOfThreadAvail:
        threadFunctionExec(thread_id,csvPath)
```

- Restore 1 table with all resources. E.g 24 threads will import 24 csv
- Why?
  - Better utilization of threads

Examples

total rows for this cluster is 134593065 (graph1)
total rows for this cluster is 164899336 (graph2)
total rows for this cluster is 135239470 (graph3)
total rows for this cluster is 129967646 (graph4)
total rows for this cluster is 141091882 (graph5)
total rows for this cluster is 91608595 (graph6)
total rows for this cluster is 131502236 (graph7)
total rows for this cluster is 81910922 (graph8)
total rows for this cluster is 59256481 (graph9)
total rows for this cluster is 66260216 (graph10)

---

1 136 329 849 rows ~ 600gb

over estimate = 164899336 * 11 = 1 813 892 696 (rows)

Backup Metrics

- with 24 threads exporting 4 millions rows / threads

[root@hdsj21msl003 ~]# time java -jar myqlPOC-0.3.0-shaded.jar

time taken for exporting in ns 44636387465

Execution time in seconds : 50

%Cpu(s): 4.0 us, 0.1 sy, 0.0 ni, 95.9 id, (we have 28 cores so 4.0%us is nothing)

- Parallel compute: 96 000 000 rows / 50 secs for 24 threads
- 1 billions rows / 96 millions ~= 12 times
- 12 * 50 = 528 secs ~= 10 minutes
  - Note I accounted for the overhead of the code

Next Step

- Install metricsManager on test server and raphael’s db server to get cpu/memory utilization when these backup job is running

Get Under 5 mins SLA

- adding more concurrency;
  - this method is very network and i/o heavy not cpu intensive,
  - we can add 44 threads , 2 threads / cpu
- add a faster storage
- Tune innodb