THE UNIVERSITY OF SYDNEY

School of Computer Science
Dr. Ying Zhou

COMP5338: Advanced Data Models **Sem. 2/2020**

Week 11: Bigtable

10.11.2020

Learning Objective and Scenario

The focus of this week's tutorial is to understand the Bigtable data model and query processing using simple example. In particular we focus on:

- The multidimensional sorted map concept and LSM tree processing
- Role of Chubby Lock Service
- Distributed read and write processing

Question 1: Multidimensional sorted map and LSM

This question assumes a Bigtable `movies` table that stores users' ratings on movies. Each user has a unique user Id and can only give one rating to a particular movie. The user may update or delete his/her rating. We store all ratings of a movie in the same row, the row key is a string representing the concatenation of a movie's title and release year. We assume the combination of movie title and release year can uniquely represent a movie. The table has a column family "r", each rating is stored in a column in this column family, with user Id as the qualifier. Below is a sequence of updates that need to be recorded in the table. Assume the table has no data at the beginning and it will not split before the end of the update sequence. Assume the table keep the latest 3 versions of a column.

- At t_0, user "u1" gave a rating of 5 to movie "m1y1". "m1y1" is the concatenation of movie title "m1" and release year y1.
- At t_1, "u1" gave a rating of 4 to movie "m2y2".
- At t_2, user "u2" gave a rating of 4 to movie "m1y1".
- At t_3, user "u2" gave a rating of 5 to movie "m3y3".
- At t_4, a minor compaction happens
- At t_5, user "u3" gave a rating of 4 to movie "m2y2".
- At t_6, user "u1" removed his/her rating for movie "m1y1".
- At t_7, user "u2" changed his/her rating for movie "m3y3" to 4.

SSTable t6
{"m1y1", "r.u1", t0} -> 5
{"m1y1", "r.u2", t2} -> 4
{"m2y2", "r.u1", t1} -> 4
{"m3y3", "r.u2", t3} -> 5

Meltable:
{"m1y1", "r.u1", t6} -> null
{"m2y2", "r.u3", t5} -> 4

- At t_8, user “u2” gave a rating of 5 for movie “m2y2”.
- At t_9, a minor compaction happens
- At t_{10}, user “u1” gave a rating of 5 to movie “m3y3”.
- At t_{11}, a merging compaction happens

Table 1 shows the summary of ratings given by users “u1”, “u2”, “u3” to movies “m1y1”, “m2y2”,“m3y3” as described in the above sequence.

SSTable t11
{“m1y1”, “r.u1”, t0} -> 5
{“m1y1”, “r.u1”, t6} -> null
{“m1y1”, “r.u2”, t2} -> 4
{“m2y2”, “r.u1”, t1} -> 4
{“m2y2”, “r.u2”, t8} -> 5
{“m2y2”, “r.u3”, t5} -> 4
{“m3y3”, “r.u1”, t10} -> 5
{“m3y3”, “r.u2”, t3} -> 5
{“m3y3”, “r.u2”, t7} -> 4

Table 1: Ratings given by three sample users to three sample movies

	u1	u2	u3
m1y1	5 (t_0) delete (t_6)	4 (t_2)	
m2y2	4 (t_1)	5 (t_8)	4 (t_5)
m3y3	5 (t_{10})	5 (t_3) 4 (t_7)	

memtable:• At t 5 , user “u3” gave a rating of 4 to movie “m2y2”.
• At t 6 , user “u1” removed his/her rating for movie “m1y1”.

Stable:t1-t4

a) What would be the content of `memtable` and `SSTable` file(s) between t_6 and t_7 ?

b) What would be the content of `memtable` and `SSTable` after t_{11} before any new update?

empty

Question 2: Bigtable cluster operation

Suppose we have a running chubby service and a Bigtable cluster with 1 master and 3 tablet servers. There is a root tablet and a METADATA tablet. The cluster currently manages five tables: `tbl1` and `tbl2` each has one tablet; `tbl3` has two tablets; `tbl4` and `tbl5` each has 3 tablets each.

10 tablets Tablets hold table data.

a) Chubby and Metadata

3 files as one file per server

1. How many files are there under the special server directory in chubby service?
2. How many rows are there in the root tablet? One row. One row of root maps to one METATABLE tablet.
3. how many rows are there in the METADATA tablet? Every tablet is defined by the key range it supports. There are 10 tablets.

METADATA tablet use key as index/ identify a route to each tablet.

b) Read and Write Path
We denote the three tablet servers as `TSR1`, `TSR2` and `TSR3`. Each server and their managed tablet are shown in table 2. For simplicity, if a table is split into a few tablets, we add a number to the table name as the identifier of individual tablet. For instance the first tablet of `tbl3` is identified as `tbl3.01`. Each tablet manages a range of keys. Since keys are globally sorted in tablets, the start or end key of tablet can be used to indicate key range of each tablet. Bigtable stores the end row key of each tablet as part of the metadata. In table 2 we show the end row key in each tablet. We

omit this information for tables that are not split. The key range of each tablet can be easily inferred using table 2 data . For instance, we can tell that row keys up to “4444” are stored in the tablet tbl3.01; Row keys between “4444” (exclusive) and “9999”(inclusive) are stored in tablet tbl3.02.

Table 2: Key ranges and location for tbl3,tbl4 and tbl5

Tablet	End row	Tablet Server
root		TS3
METADATA		TS2
tbl1		TS1
tbl2		TS3
tbl3.01	“4444”	TS2
tbl3.02	“9999”	TS1
tbl4.01	“hhhh”	TS3
tbl4.02	“ssss”	TS1
tbl4.03	“zzzz”	TS2
tbl5.01	“JJJJ”	TS2
tbl5.02	“TTTT”	TS1
tbl5.03	“ZZZZ”	TS3

Answer the following questions:

1. Which one of the following statements is true?

   A TS1 does not store files belonging to tbl2. FALSE, it may store files belonging to tb12 later down the line. At the moment it may not store locally but in time it may

   B The data in tbl5 is stored in three SSTable files. A tablet contains multiple SSTables thus FALSE

   C TS2 manages 4 tablets.

   D All of the above.

2. A new client wants to read row “comp” from tbl4, Describe in detail the sequence of steps taken to carry out this read operation. In particular, describe in each step which server/service is contacted for what information.

3. Another client wants to update row “OLYM” from table tbl5.Describe in detail the sequence of steps take to carry out this write operation.

c) Fault Tolerance

Now suppose the network connection between TS1 and Chubby service is broken. How would the cluster react to this situation?

2.2
Step 1: contact chubby service for root table, returns TS3
Step 2: contact TS3 to get location of METADATA table. Returns TS2
Step 3: contact TS2 to get location of tablet that has “comp” in tbl14. Tbl14 is stored on TS1, TS2 and TS3. As TS3 contains the keys up to “hhhh” then “comp” is stored in this server. Returns T3

Step 4: client sends read request to TS3, which searches memtable as well as SSTable files to find the required data.

2.3
Step1:Step2:
Step 3: Contact TS2 to get location of tablet that’s “OLYM” from table tb15. TS1 contains tb15.02 which has the keys from “jjjj” up to “ttttt” which “OLYM” is within. TS2 returns the sever that manages tb15.02 which is TS1.
Step 4: client sends write request to TS1, which will append operation to commit log and insert into meltable.

TS1 is still able to communicate to master but can not extend its lock leads with the chubby server. It can talk to master but can’t keep its lock.
The master attempts to acquire the lock of TS1 file on Chubby and delete the file after the lock is acquired.
This means that TS1 would not be considered as a tablet server in the cluster.
- The master scans the EMTADATA table to reassign tablets managed by TS1 to other two tablet servers. Currently, tb1, tb3/02, tb4.02 and tb5.02 are managed by TS1. They will be re-assigned to TS2 and TS3. It is likely that each may get two tablets.
-The reassignment happens by master sending load tablet request to the tablet server. Upon receiving the load tablet requires, the tablet server will perform the standard tablet recovery steps.