Distributed Systems

2018 Pre-exam 3 review
Selected questions from past exams

David Domingo

Paul Krzyzanowski

Rutgers University

Fall 2018

Question 1 (Bigtable)

What is an *SSTable* in Bigtable?

It is the internal file format used to store Bigtable data.

It maps *keys* to *values*, both of which can be arbitrary byte strings.

See *Bigtable:* section 4, page 3

Question 2 (Bigtable)

What is a *memtable* in Bigtable and how and when does it become an SSTable?

It's memory-resident recent table data

Write operations are logged and then written to a memory-based table of recent changes, called a memtable.

Once it reaches a certain size, it gets written as an SSTable

"As write operations execute, the size of the memtable increases. When the memtable size reaches a threshold, the memtable is frozen, a new memtable is created, and the frozen memtable is converted to an SSTable and written to GFS." (section 5.3:)

Multiple SSTables comprise a tablet. The METADATA table contains a list of SSTables that form a tablet. Multiple tablets comprise Bigtable.

See *Bigtable:* section 5.4

Question 3 (MapReduce)

How does an *Input Split* differ from an HDFS block?

Input Splits allow MapReduce to be told where the record boundaries are in the input so that a mapper won't be left with a partial record or start with a partial record.

An HDFS block is just a 128 MB chunk of a file.

Question 4 (MapReduce)

What two operations take place during the shuffle phase of MapReduce?

Sort: Each mapper sorts its (*key, value*) data by key.

Merge: Each reducer contacts each mapper for all sorted (*key, value*) sets for that reducer and merges them.

Question 5 (Spanner)

What does spanner mean by *external consistency*?

External consistency is when the commit order is the same as users actually see the transactions executed with respect to wall-clock time.

Question 6 (Spanner)

What is a *commit wait* in Spanner?

- Commit wait is having a transaction wait until the transaction’s commit timestamp is definitely in the past prior to committing and releasing locks
- This ensures that any future transaction that accesses any of that data will get a timestamp that is greater than the previous transaction.
- A transaction timestamp is set to TT.now().earliest

  = the earliest possible time we know it can be currently
- On a commit wait, the transaction waits until the current earliest time is greater than TT.now().latest

  = the latest possible time we know it can be currently

Some past exam questions

Fall 2016: Question 2

You have access to a file of class enrollment lists. Each line contains *{course_number, student_id}*.

Explain how you would use MapReduce to get information on how many classes students take.

For instance, you may discover that 1,495 students are enrolled in 6 courses; 13,077 students are enrolled in 5 courses; 14,946 students are enrolled in 4 courses; and 4,484 students are enrolled in 3 courses.

Explain each map and reduce operation. You may use pseudocode and assume that functions such as sum and count exist. Be sure to state the inputs & outputs of each step.
Hint: you may need more than one iteration

Fall 2016: Question 2 (cont.)

You have access to a file of class enrollment lists. Each line contains *{course_number, student_id}*.

First MapReduce: find # of courses each student takes

```
Map_1:
        input: { course_number, student_id }
        output: { key=student_id, 1 }
Reduce_1:
        input: { student_id, courses[] }
        output: { student_id, sum(courses) }
```

1 for each course per student

We can also output { 1, sum(courses) }

Second MapReduce: find # of students that take each course count

```
Map_2:
        input: { student_id, course_count }
        output: { course_count, 1 }
Reduce_2:
        intput: { course_count, students[] }
        output: { course_count, count(students[]) }
```

1 for each student with course_count courses

Fall 2016: Question 2 (cont.)

Input

213	130972375
416	162692062
416	534744968
416	021693896
417	162692062
417	130972375
519	130972375
...	

Output from map

key=130972375,	1
key=162692062	1
key=534744968	1
key=021693896	1
key=162692062	1
key=130972375	1
key=130972375	1
…	

Input to reduce

130972375	1, 1, 1
162692062	1, 1
534744968	1, 1, 1, 1
021693896	1, 1, 1, 1
…	

Output from reduce

130972375	3
162692062	2
534744968	4
021693896	4
…	

Student 130972375 takes 3 classes

Fall 2016: Question 2 (cont.)

Input to map =
output from reduce

130972375 3
162692062 2
534744968 4
021693896 4
…

Output from map

key=3, 1
key=2, 1
key=4, 1
key=4, 1
…

"one student taking 2 courses"

"one student taking 4 courses"

"one student taking 4 courses"

Input to reduce

2, {1, 1, 1, 1, ... }
3, {1, 1, 1, 1, 1, … }
4, {1, 1, 1, 1, 1, 1, … }
…

Output from reduce

2, 1622
3, 4484
4, 14946
…

Fall 2017: Question 1

The core task of the *user's map function* within a *map* worker in a MapReduce framework is to:

(a) Determine which reduce worker should process which key.
(b) Split the input data into shards.
(c) Parse input data and create key, value tuples.
(d) All of the above.

Framework – splits data

Partitioning function – determines which reduce worker handles a key

Fall 2017: Question 3

Reduce workers in MapReduce can start working:

(a) In parallel when the map workers start.

(b) When at least one map worker starts to generate data.

(c) When at least one map worker has processed all its input.

(d) When every single map worker has completed its task

All <key, value> sets must be generated before ***any*** reducer can start

Fall 2017: Question 4

Bigtable's *multidimensional* property refers to the fact that:

(a) Bigtable stores versioned data within rows and columns.

(b) A table is actually composed of an arbitrary number of tablets.

(c) A multi-level storage structure is used: memtable, SSTable, tablet, and table.

(d) Each cell in a table can also be a table and, recursively, cells within that table can be tables.

---

b & c: true but don't answer the question

d. Not supported in Bigtable

Fall 2016: Question 3

How does Spanner provide consistent lock-free reads of lots of data even if other transactions are modifying some of that data during the read?

Spanner stores multiple timestamped versions in each field.

Snapshot reads allow reading of data whose
    version ≤ transaction start timestamp

Fall 2015: Question 2

Explain the role of dynamic DNS in a content delivery network (CDN).

- It directs the client to a caching server operated by the CDN instead of to the origin server
- This will generally be the closest active server
- DDNS may use load balancing to give addresses if different servers

Bad answers:

- *It gives the most efficient route.*
  This isn't accurate since DNS does not dictate routes; it just gives addresses. We hope that the closest address will be the one with the most efficient route.
- *Server that contains the content.*
  DNS doesn't know what the content query will be: it just gets the domain name.

Fall 2017: Question 14

Pregel's combiners:

(a) Reduce the number messages from the same processor that are targeted to the same destination.

(b) Manage global state.

(c) Merge multiple vertices into one vertex.

(d) Merge multiple edges into one edge.

Combiner = optional function to consolidate messages to the same vertex

Aggregator = global state

Fall 2017: Question 15

In Spark, a *Resilient Distributed Dataset*, or RDD, is:

(a) A distributed collection of objects that is modified by each transformation.

(b) An immutable distributed collection of objects representing original data or the output of a transformation.

(c) The original input data that will be processed by Spark and is replicated onto multiple servers.

(d) The output data generated by a Spark action.

---

a. An RDD is immutable = never modified

c. An RDD can be original data or the output of a transformation

d. Only actions produce final data. Prior to that we have transformations.

Fall 2017: Question 20

A *clustered file system* differs from a distributed file system in that:

(a) Multiple computers access the same physical storage device.

(b) Data may be distributed among multiple computers.

(c) Data is replicated across storage devices on multiple computers for fault tolerance.

(d) It provides services only over a local area network.

2015 Question 19

In contrast to a shared-nothing cluster, a *shared-disk* cluster relies on a:

a) Quorum service.
b) Heartbeat network.
c) Cluster membership service.
d) Distributed lock manager (DLM).

- Multiple machines may issue read/write requests for the same block at the same time. A DLM will ensure mutual exclusion.

Fall 2017: Question 22

Fencing is used to:

(a) Provide a trusted path for nodes to communicate on a LAN.

(b) Isolate a computing node from other nodes.

(c) Monitor whether cluster members are alive.

(d) Establish a quorum among cluster members.

Fencing shuts off or isolates components that may be misbehaving.

Fall 2014 - Question 4

Alice has Bob's X.509 digital certificate. She validated it to ensure that it is legitimate.
How does she now use it to establish a secure communication channel so she and Bob can exchange encrypted messages?

We’re *not* asking Alice to validate Bob – just to communicate securely.

By possessing Bob’s certificate, Alice has his public key.

1. Alice creates a random session key S.
2. Alice encrypts S with Bob’s public key in his certificate.
3. Alice sends the encrypted key to Bob.
4. Bob decrypts the session key using his private key.
5. Alice & Bob now have a shared key and can communicate.

Fall 2014 - Question 4 – Discussion

Alice has Bob's X.509 digital certificate. She validated it to ensure that it is legitimate.
How does she now use it to establish a secure communication channel so she and Bob can exchange encrypted messages?

This is not the question, but…
If Alice first wanted to validate that she's talking with Bob:

1. Alice generates a random string (nonce) and sends it to Bob.
2. Bob encrypts it with his private key and sends the result to Alice.
3. Alice decrypts the received message using Bob's public key (in his certificate). If the result matches the nonce, she is convinced.

2015 Question 22

In contrast to symmetric cryptography, *public key cryptography*:

a) Solves the problem of transmitting a key securely.
b) Is usually much faster than symmetric cryptography.
c) Is designed for group communication.
d) Is useful for digital signatures, not encryption.

- Symmetric cryptography requires both parties knowing a shared secret key
- This has to be transmitted out of band or encrypted
  - Only way to encrypt is using a trusted third party
  - However, this requires getting the trusted third party to know each user's key

The End