MSDS 7330 File Organization and Database Management Final Exam

Name: Cory Nichols

Final Exam: Directions

This is a final exam for MSDS 7330, File Organization and Database Management. This document contains the questions for the exam. For your answers, create a Word document that clearly identifies every question number and your answer to that question. Name the Word file containing your answers ‘yourLastNameMSDS7330ExamFinal.doc’. For example, my Word file would have the name EngelsMSDS7330ExamFinal.doc.

Answer each question fully and completely. Show all of your work and state your assumptions where appropriate. Each answer is worth an equal number of points. For each multiple choice question, record the one letter of your chosen answer (1/2 the points) and write 2-5 sentences explaining why your chosen answer is correct (1/2 the points). Note that one sentence explanations will receive 0 points even if they are correct.

The questions may have hints embedded within them regarding the answer. Follow these hints as appropriate for full points.

Collaboration is expected and encouraged; however, each student must hand in their own exam. To the greatest extent possible, answers should not be copied but, instead, should be written in your own words. Copying answers from anywhere is plagiarism, this includes copying text directly from the textbook. Do not copy answers. Always use your own words. Directly under each question list all persons with whom you collaborated and list all resources used in arriving at your answer. Resources include but are not limited to the textbook used for this course, papers read on the topic, and Google search results. Don’t forget to place your name in the Word document itself.

**Final Exam: Questions**

**1) In traditional RDBMS, a table may contain which of the following?**

a) Complex data structures

b) Arrays

 c) Embedded classes

d) All of the above

e) None of the above

**ANSWER:** E – None of the above.

An RDBMS table is restricted compared to NoSQL. Tables contain mostly atomic values that are not in array format. For instance, an array would be stored in multiple attributes or even separate relations. Further, RDBMS’s do not support embedded classes or complex data structures such as nested documents.

**2) Which of the following is a potential driver for the development and/or adoption of a NoSQL database?**

a) Simplicity in the database design (e.g., the NoSQL database is simpler than the corresponding SQL database).

b) Minimize the number of translations from how the data is stored to how the data is consumed.

c) Flexibility in the database design.

d) All of the above. [Hint: a description for this answer would explain the benefit for each of the above possible  answers.]

e) None of the above. [Hint: a description for this answer would explain how SQL provides each of the above possible  answers.]

**ANSWER:** D – All of the above

NoSQL databases allow for great flexibility in schemas and data types. A schema does not have to be identified prior to creating a database in MongoDB, for instance. Further, NoSQL is joinless. Meaning that users can consume the data much more easily. Database design is also simple. No E:R models are required! Further, schemas are dynamic and designs can change more easily than in the relational model.

**3) In the paper “MapReduce: Simplified Data Processing on Large Clusters,” by Jeffrey Dean and Sanjay Ghemawat,  published in the Communications of the ACM, Jan 2008, the concept of performing a map function followed by a reduce function is introduced. The computations take *what* as input and produce *what* as output?**

 a) Tables, Tables

b) Key/Value pairs, Key/Value pairs

c) Documents, Tables

d) XML, XML.

**ANSWER:** B – Key/Value pairs, Key/Value pairs

Map functions take a key/value pair, which generate a set of intermediate key value pairs. From there, reduce functions merge the intermediate values associated with the same intermediate key. These functions combine data, while working across distributed systems in order to handle GBs or TBs of data that are too large to fit in memory.

4) **In the paper “MapReduce: Simplified Data Processing on Large Clusters,” by Jeffrey Dean and Sanjay Ghemawat, published in the Communications of the ACM, Jan 2008, the concept of performing a map function followed by a reduce function is introduced. Which of the following is true of the way in which the MapReduce operations work?**

a) Each instance of the map function receives the complete set of data.

b) Each instance of the MapReduce functionality randomly chooses whether to perform the map function or the reduce  function.

c) Each instance of the reduce function receives all of the intermediate data from every map function.

d) All of the above. [Hint: a description for this answer would explain the basic operation for each of the above  possible answers.]

e) None of the above. [Hint: a description for this answer would provide the correct operation and explain the basic  of this operation for each of the above possible answers.]

**ANSWER:** C – Each instance of the reduce function receives all of the intermediate data from every map function.

Each reduce function receives all intermediate key value pairs from the map function. The reduce function then produces a compressed, sorted key value pair. Map functions work with partitions of the complete data set and store their intermediate results on local disk. Further, the master chooses what workers perform map or reduced functions based on their availability, not at random. The article explicitly states that a reduce worker will read all intermediate data.

**5) In the paper “MapReduce: Simplified Data Processing on Large Clusters,” by Jeffrey Dean and Sanjay Ghemawat,  published in the Communications of the ACM, Jan 2008, the concept of performing a map function followed by a reduce function is introduced. Which of the following statements is true about the intermediate data generated by the map worker functions and the reduce worker functions? [Hint: If a single answer (a), (b) or (c) is chosen, explain why the chosen answer is correct and the other two are incorrect.]**

a) The map intermediate results are stored on local machines.

b) The reduce worker results are stored on local machines.

c) The map intermediate results are never combined or manipulated before being input to the reduce worker function.

d) All of the above. [Hint: a description for this answer would explain the basic operation for the map and reduce  worker functions.]

e) None of the above. [Hint: a description for this answer would provide the correct operation and explain the basic  operation for the map and reduce worker functions.]

**ANSWER:** A – The map intermediate results are stored on local machines

Completed map tasks are stored on local disks. Reduce tasks are stored in a global file system. Further, map intermediate results are simply the key, value pairs that are not combined. Reduce functions handle the combining and consolidation in a mapreduce environment.

**6) In the paper “MapReduce: Simplified Data Processing on Large Clusters,” by Jeffrey Dean and Sanjay Ghemawat,  published in the Communications of the ACM, Jan 2008, the concept of performing a map function followed by a reduce function is introduced. Which of the following are identified as advantages of MapReduce? [Hint: If a single answer (a), (b) or (c) is chosen, explain why the chosen answer is correct and the other two are incorrect.]**

a) MapReduce allows programmers with no parallel system experience to exploit large amounts of resources easily.

b) MapReduce makes it possible to write simple programs that run efficiently on large numbers of machines.

c) MapReduce allows for a large variety of problems to be easily expressible as MapReduce problems.

d) All of the above. [Hint: a description for this answer would explain why each of the answers above provides value.]

e) None of the above. [Hint: a description for this answer would explain why each of the answers above is not a value  provided by MapReduce.]

**ANSWER:** D – All of the above

MapReduce functionality is based on large, distributed systems with workers assigned to the map and reduce functions. These distributed system resources work in conjunction when a MapReduce function is called to manipulate extremely large amounts of data across multiple systems with simple programs. Because of this flexibility, many different types of large data problems such as machine learning and logging can be handled via MapReduce.

**7) In the paper “BigTable: A Distributed Storage System for Structured Data,” by Chang *et al.*, published in OSDI 2006,  a distributed storage system for managing structured data that scales to petabytes is presented. The goals for BigTable include which of the following?**

 a) Scalability

 b) High performance

 c) High availability

d) Wide applicability

e) All of the above. [Hint: a description of this answer would explain each of the answers a-d above.]

f) None of the above. [Hint: a description for this answer would provide an explanation of the goals for BigTable.]

**ANSWER:** E – All of the above

Bigtable was created to manage large quantities of structured data over a distributed storage system. It does not support a full relational model, instead it provides for a simple data model allowing dynamic control over data layout and format. It is in use for over 60 projects at Google.

**8)  In the paper “BigTable: A Distributed Storage System for Structured Data,” by Chang *et al.*, published in OSDI 2006, a distributed storage system for managing structured data that scales to petabytes is presented. Which of the following  best describes BigTable?**

a) BigTable provides a scalable implementation of a traditional relational database.

b) BigTable provides a complex data model that supports dynamic control over data layout and format.

c) BigTable provides a simple data model that supports dynamic control over data layout and format.

d) BigTable provides a simple interface to a traditional relational database that supports dynamic control over data  layout and format.

**ANSWER:** C - Simple data model that supports dynamic control over data layout and format.

Bigtable’s data model is distributed with sparse indices on row and column keys as well as timestamps. The index is an uninterpreted array of bytes. Rows are made up of arbitrary strings written and read atomically. Columns are grouped into families – which are usually the same type and versioning is handled via timestamps as 64 bit INT values.

**9) In the paper “BigTable: A Distributed Storage System for Structured Data,” by Chang *et al.*, published in OSDI 2006, a distributed storage system for managing structured data that scales to petabytes is presented. BigTable provides which of the following traditional RDBMS functionalities?**

a) Atomic write operations.

b) Access control rights.

c) Atomic read operations.

d) All of the above. [Hint: a description for this answer would explain each of the answers a-d above.]

e) None of the above. [Hint: a description for this answer would provide two examples, and explanations of what  functionalities are provided by BigTable.]

**ANSWER:** D – All of the above

Every read or write of data under a single row is atomic in bigtable. Single row transactions are fully supported under a single row key. Across row keys, however, transactions are not atomic. Access control rights are handled via columns in BigTable using column families, which should be kept to a minimal set.

**10) In the paper “BigTable: A Distributed Storage System for Structured Data,” by Chang *et al.*, published in OSDI 2006, a  distributed storage system for managing structured data that scales to petabytes is presented. BigTable utilizes a hierarchy analogous to that of a B+ tree. How many levels are in this hierarchy? [Hint: explain each level in the hierarchy.]**

a) 2

b) 3

c) 4

d) 5

**ANSWER: B –** Three

The first level is ‘chubby’ – a library that contains the location of the root tablet (server). The ‘root tablet’ is a massive server with a metadata table containing the location of all of the tablet servers. The second level is the population of metadata tablets containing the location of a set of user tablets. The user tablets make up the third level. The root tablet is technically the first server in the metadata tablets, so it is not treated as a level.

**11) Which of the following data is least suited to a traditional RDBMS database?**

a) Book title

b) Multiple authors

 c) Publication date

d) An abstract spanning several paragraphs.

**ANSWER:** B – Multiple authors

Multiple authors is an array. This would not be efficiently represented in a relational database without creating individual attributes for each author. An abstract of several pargraphs could easily be stored in a relational database as a CLOB or TEXT object, which can then be searched on. Because NoSQL databases like MongoDB support complex object types, multiple authors would be more suitable for a document or key:value pair database. Book titles and publication dates are standard storage for an RDBMS.

**12) In MongoDB, a table (called a collection) may contain which of the following?**

a) Complex data structures

b) Arrays.

c) Embedded classes

d) All of the above

e) None of the above

**ANSWER: D** – All of the above

MongoDB can store complex data structures, arrays and embedded classes. Complex objects like a dictionary can easily be stored as a complex data structure, along with a list of values (an array). Embedded classes and sub documents are easily stored within a document in MongoDB.

**13) The paper “Dynamo: Amazon’s Highly Available Key-value Store,”by DeCandia *etal.*, publishedinSOSP2007,presents a highly available key-value storage system used by Amazon.com. Scalability and reliability are the primary drivers for the design and implementation of Dynamo. To this end, hardware and network failures are treated as which of the following?**

a) The normal case

b) Rarely occurring special case.

 c) Often occurring special case.

d) They are not explicitly considered in the design.

**ANSWER: A** – The normal case.

The normal case is that hardware and network failures are normal at Amazon. Millions of customers interact with Amazon.com daily. The traffic is distributed across thousands of servers and hundreds of services. Amazon’s primary goal is reliability and availability. Data should be able to be read and written even when disks fail, which requires horizontal scalability, sharding and replication on a massive scale.

**14) The paper “Dynamo: Amazon’s Highly Available Key-value Store,”by DeCandia *etal.*, published in SOSP2007,presents a highly available key-value storage system used by Amazon.com. Dynamo provides only what kind of access to the data store?**

a) Candidate key

b) Primary key

c) Superkey

d) Foreign key

**ANSWER: B** – Primary Key

Amazon Dynamo DB uses a primary key approach to access the data store. A key uniquely identifies data items with no need for a relational schema. Dynamo targets object storage of 1MB or less typically. This works well with shopping cart data, which Amazon relies on for the majority of their revenue.

**15) The paper “Dynamo: Amazon’s Highly Available Key-value Store,” by DeCandia *etal.*, published in SOSP 2007, presents a highly available key-value storage system used by Amazon.com. Dynamo relaxes which database property to maintain high availability?**

 a) Isolation

b) Durability

c) Atomicity

d) Consistency

**ANSWER: D** – Consistency

Dynamo DB relaxes consistency in order to achieve high availability. Amazon follows an eventually consistent model with Dynamo DB. ACID properties oftentimes prevent highly available data infrastructure, which is the primary focus of Amazon’s business. Lost revenue and customers would occur in magnitudes if customers had to wait for server availability in order to ensure consistency.

**16) The paper “Dynamo: Amazon’s Highly Available Key value Store,” by DeCandia *etal.*, published in SOSP 2007, presents a highly available key-value storage system used by Amazon.com. Dynamo achieves its performance by utilizing which of the following techniques?**

 a) Uniform data distribution through consistent hashing

b) Trade off durability guarantees for performance

c) Uniformly assigning requests to nodes

d) All of the above.

e) None of the above.

**ANSWER: D** – All of the above

Clients are able to trade off durability to manage extremely high performance SLAs. Objects are read from buffers instead of the storage engine. Keys are uniformly distributed through consistent hashing to achieve uniform load distribution and reduce storage sizes in individual nodes, which would slow down access to data. Further, requests are uniformly assigned to nodes. Specifically read and write operations are optimized to use the fastest read-responsive node as the write coordinator, increasing performance at the 99.9 percentile.

**17) Brewer’s CAP Conjecture states:**

a) It is possible for a distributed database to achieve consistency, availability and partition tolerance simultaneously.

b) It is possible for a distributed database to achieve at most one of consistency, availability and partition tolerance  simultaneously.

c) It is impossible for a distributed database to achieve consistency, availability and partition tolerance, or any  combination of the three, simultaneously.

d) It is impossible for a distributed database to achieve consistency, availability and partition tolerance simultaneously.

**ANSWER:** D – Distributed databases cannot achieve all three of CAP Theorem simultaneously.

The CAP Theorem states that satisfying consistency, availability and partition tolerance simultaneously is impossible. A business or developer must choose which two are the most important for the application or objective at hand. For instance, Amazon’s Dynamo database sacrifices consistency in the CAP theorem. If an e-commerce company’s website is not available and risk is not hedged with shads and replications, that company will likely not stay in business.

**18) NoSQL databases are architecturally different from relational database because:**

a) they represent relational data in a different manner.

b) they are designed to reap the read and write performance benefits of partition tolerance (horizontal scaling) while  leaving either consistency or availability up for negotiation.

c) they are designed to reap the certainty benefits of consistency while leaving either partition tolerance or availability  up for negotiation.

d) they are designed to reap the access benefits of availability while leaving either partition tolerance or consistency  up for negotiation.

**ANSWER:** B – Horizontal Scaling

NoSQL databases can easily be scaled horizontally. Relational databases are normally scaled vertically on large mainframes, while NoSQL databases are scaled horizontally with commodity hardware. Many NoSQL databases may represent relational data in an embedded document format, for instance with MongoDB.

**19) MongoDB:**

 a) Doesn’t provide ACID guarantees over a series of operations.

b) Doesn’t have the equivalent of an RDBMs’ BEGIN, COMMIT and ROLLBACK semantics

c) Supports atomic, durable updates on individual documents and consistent reads.

d) All of the above

 e) None of the above.

**ANSWER:** D – All of the above

MongoDB cannot ensure atomicity on multiple operation transactions across documents or collections. However, within a single document, MongoDB updates are atomic and power-outage resistant. Further, MongoDB does not support transactional integrity using BEGIN, COMMIT and ROLLBACK. If transactional integrity is required, a RDBMS is the best choice.

**20) Which of the following statements is true of MongoDB operations?**

a) if you have a multi-operation transaction that decrements from one property and increments another (on a single  document), then MongoDB can ensure atomicity and durability in this case.

b) if you have a multi-operation transaction that updates properties on separate or multiple documents, then MongoDB  cannot ensure atomicity or durability in this case.

c) if you have a multi-operation transaction on a single document, locking strategies are used to ensure strong  consistency at the potential cost of high availability.

d) All of the above.

e) None of the above.

**ANSWER:** D – All of the above

MongoDB is not atomic across multiple documents. However, on a single document, MongoDB ensures atomicity and durability of data. When working across multiple documents, data can become stale and durability may be an issue. A transaction would not be able to fully commit if working with multiple documents and a power outage is experienced. Mongo also uses locking to ensure data consistency while sacrificing availability.

**21) Which of the following types of databases is most suitable for banking transactions? [Hint: in your explanation describe an example scenario that illustrates why your chosen type works.]**

 a) Relational type

b) Document type

c) Graph type

d) Key-Value type

**ANSWER:** A – Relational Type

Relational type databases are ACID compliant. NoSQL databases sacrifice ACID aspects in favor of various other features. Because data consistency, atomicity, isolation and durability are all absolutely critical in banking, we’d use a relational database for this application.

**22) The Cassandra database operates best under what relative read-write occurrences?**

a) Equal numbers of reads and writes.

b) More writes than reads.

c) More reads than writes.

 d) All of the above.

e) None of the above.

**ANSWER: B** – More writes than reads.

Writes involve less I/O while reads involve more I/O overhead. Writes are written to a commit log and a memtable (memory)  in order. This sequential nature allows writes to perform faster than reads, which have a less structured process.

**23) MongoDB is designed to operate with data stored across multiple servers. The process of distributing different data across multiple servers is referred to as?**

 a) Sharding

b) Replication

c) Copying

d) Splitting

**ANSWER:** A – Sharding

Sharding allows data to be distributed across multiple servers. Replication provides backup and data synchronization to distributed nodes. Copying is a form of replication, really it’s just made up, while splitting is analgous to sharding, but this is also a made up term.

**24) Which of the following companies developed Cassandra?**

a) Twitter

b) Google

c) LinkedIn

d) Facebook

**ANSWER:** D – Facebook

Facebook. Mark Zuckerberg didn’t steal this one. It is open source and highly scalable. Cassandra’s data model is a row store.

**25) List five 1-2 word advantages of NoSQL databases (generally speaking) over relational databases, and list five 1-2 word disadvantages of NoSQL databases (generally speaking) compared to relational databases. Provide a one to two sentence explanation for each advantage and for each disadvantage.**

**NoSQL Advantages Over Relational:**

1. Dynamic Schema
   1. No E:R diagrams required. Schema can be changed ad-hoc.
2. Horizontal Scaling
   1. Workloads and replication can be spread across multiple servers, multiple locations.
3. Cost Effective
   1. Most (if not all) NoSQL databases are open source. Further, it is often cheaper to scale horizontally on commodity servers than add mainframes.
4. Natural for Programmers
   1. Developers can easily pick up NoSQL syntax. Not much up front language learning required.
5. Require less management
   1. Simpler data models, automatic repairs and data distributions require less management than bulky schemas and relation maintenance in an RDBMS.

**NoSQL Disadvantages vs Relational:**

1. Less Generic
   1. NoSQL is like a toolbox. Different databases solves different problems while relational DBs solve more general problems.
2. Non-ACID compliant
   1. NoSQL databases often sacrifice consistency and isolation in favor of availability and partition tolerance.
3. Less Mature
   1. Community support is not as mature and bolt on tools aren’t as common. SQL is known by far more people than NoSQL syntax.
4. Not BI-Friendly
   1. Traditional data analysts are not programmers and will have a tough time learning NoSQL queries. Data not structured to easily retrieve and analyze.
5. Open Source
   1. There are not many standards established for NoSQL databases. This makes NoSQL solutions more difficult to install and interact with more established applications.