Name: Charitha Kammari

Ucid: ck369

CS 643, Cloud Computing – Homework 3

1. (3 points) What is the advantage of using the record append function instead of the write function in GFS? Are GFS replicas guaranteed to have the same content if only record append functions are used to write data? Justify your answers.

Answer: GFS means Google File System; this essay talks about the advantages of the record append function instead of the write function.

Advantages of using the record append function:
One advantage of the record append function is that it allows data to be added to files quickly, especially in situations when there are many writes. While the record add function appends data to a file at its end without requiring an offset, the write method requires one in order to write data.It can manage numerous append functions in a situation where clients are simultaneously adding entries to the same file because it is built to handle concurrent appends. By employing the "record locks" technique, it adds them in the order that they were written.

If data is written using solely record add functions, there is no guarantee that GFS copies will have the identical content. Records can be appended to a file in the desired sequence with the record append function. The number of replicas of a specific piece of data that should be stored is determined by the master node. According to the specified replication factor, the master schedules recurring "chunk re-replication" operations to add or remove replicas. The replication procedure may take longer than expected because to things like network latency and system load. In conclusion, the record add function may be superior to the write function; but, relying solely on the record function for data writing may result in occasional delays. The GFS uses extra safeguards, such as routine inspections.

2. (3 points) Give an example of a situation that causes a Bigtable tablet server to shut down. Why must the server shut down in this situation?

Answer: Structured data is handled by a bigtable tablet server. The technology in question is a distributed storage system that has the capacity to manage substantial volumes of data throughout several "tablet servers."
Let's use an example to talk about what happens when a Bigtable tablet server turns off. Tablet servers are in charge of keeping and delivering data for individual tablets while the server is managing a high volume of requests. Due to their limited memory, the tablet servers will not be able to process every request that comes in. The tablet server shuts down because it does not have enough RAM to process requests. Issues including cluster instability, performance deterioration, and data loss could result from this circumstance.

When this happens, servers must crash since the server cannot fulfill requests for data without memory. The entire contents of the tablet may be corrupted if the server isn't turned down. Increasing the server's memory, decreasing the number of processes, and improving the efficiency of the processes that are already operating on the server are some possible solutions to the server shutdown problem.
A hardware malfunction or a leased tablet server are two other scenarios in which a server can force a Bigtable tablet server to terminate.

3. (4 points) Describe two problems associated with data partitioning in distributed systems. Also, describe the solution provided by Dynamo for both problems.

Answer: Data partitioning is a technique that helps with query processing performance and data management by dividing data among several tables, systems, or sites. Data can be divided into many distinct categories, and the methods used to divide it vary based on the type of data. Let's talk about the two most popular approaches to data partitioning. They are divided vertically and horizontally.

List-based partitioning and hash-based partitioning are the two categories of partitioning techniques. Data skew and load imbalance are two issues related to data partitioning.

Data Skew:
The data skew occurs when a single partition or node in a distributed system receives an unusually high amount of data or traffic. This could lead to availability problems like unreachable partitions and performance problems like latency and throughput.

Dynamo Solution:
To solve the problem of data skew, Dynamo presents a technique called "Virtual Nodes". Using dynamic mapping, which divides each physical partition into several virtual nodes, Dynamo uses a method of key association that is not statically associated with a single physical partition or shard. Each virtual node has a somewhat smaller key range under its supervision.

Load Imbalance:
When the workload is not split evenly among the storage nodes in a distributed system, load imbalance occurs. While certain nodes may be underutilized, others may be swamped with requests. This could lead to decreased system effectiveness, greater latency, and worse performance.

Dynamo Solution:
To handle load imbalance, Dynamo employs a decentralized, flexible data placement technique. It uses a dynamic load-balancing technique that adjusts data location to enhance load distribution by continuously evaluating workload. When a node becomes overloaded, Dynamo's load

balancing algorithm can transfer data from an overloaded storage node to an underutilized node, therefore dispersing the workload. This approach will enable the system to maintain a more equitable allocation of requests and data among the storage nodes while also allowing it to adjust to changing traffic patterns.