

Data storage investigation

by Jian He

Submission date: 19-Jan-2023 07:00PM (UTC+0000)

Submission ID: 195643121

File name: 40881_Jian_He_Data_storage_investigation_474125_651877196.docx (29.57K)

Word count: 3648

Character count: 20020

Abstract

With the continuous development of information technology, new database application fields are constantly produced. In the face of business requirements such as big data and high concurrency, traditional relational databases cannot meet the needs. So the non-relational databases are also created. This paper first introduces the contents and characteristics of four relational databases and compares them with relational databases. Then this paper expounds the application field of relational database and non-relational database respectively, and draws relevant conclusions. Finally, Mysql and MongoDB respectively give the database design scheme, in order to better meet the requirements of Abertawe Entertainment company.

1 The description of several types of NoSQL databases

1.1 The introduction of NoSQL databases

In the current era of information explosion, the data is growing rapidly every day. To store and manage so much data at this time poses increasing challenges to traditional databases. The representatives in the traditional relational databases are Mysql, Oracle, SQL server, etc. In the face of a large amount of data is more difficult, especially for the flexible and free data storage format is more helpless. Hence the concept of Nosql, which is designed for unstructured data and cluster environments. It was first proposed by a software developer named Johan Oskarsson. In 2009, Johan Oskarsson first introduced the Nosql concept at a technology exchange conference in San Francisco, and for several years thereafter, thanks to Nosql.

With its excellent performance and scalability, Nosql soon was rapidly developed and promoted. So what is itNosql, what is the difference between it and the traditional relational database, currently people will split the Nosql into No

And Sql to understand, interpreted as not only sql. Literally, Nosql is a non-traditional relational database, or Nosql is relative to the traditional SQL, in fact, the correct understanding of Nosql should be NO RDBMS, that is, a non-relational database management system. They are different from the traditional relational structure database system, and are developed in order to more easily expand the cluster environment.

1.2 The theoretical basis of the NoSQL databases

No SQL Database refers to the non-relational database, No SQL, also called Not Only SQL, is a general term for the database management system different from the traditional relational database. At present, scholars at home and abroad agree that the three theories of CAP theorem, BASE theorem and final consistency theory form the

theoretical basis of No SQL database. This section will mainly explain these three theories.

1.2.1 CAP theory

At the ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing meeting, Professor Eric Brewer of the University of California first proposed the CAP theorem, where the letters C, A, and P respectively represent the following three characteristics of the distributed system:

(1) Strong consistency (Consistency): After the system completes an operation, the state of the system remains consistent.

This distributed system is considered to have strong consistency, after any update, all users shall read the latest correct values;

(2) Availability (Availability): In a distributed system, any operation must be given in the system. If such a system is considered available, and the response can be fed back make the success and failure.

(3) Tolerance (Partition Tolerance): Partition fault tolerance means that the distributed system can still process user requests as usual when the network is divided into multiple blocking areas due to failure. The CAP theorem states that a distributed system cannot meet the three requirements of consistency, availability and partition fault tolerance, but at most two at once. Therefore, the system designer must make a trade-off when designing the system and select the two conditions.

1.2.2 The BASE Theorem

Based on the CAP theorem, consistency, availability, and partition fault tolerance cannot be met simultaneously. But, for systems that need to deal with big data need more availability and partition fault tolerance. So these systems are not available to meet the ACID properties for the transaction. A transaction is an inseparable sequence of user-defined operations that are either all not executed or all executed.

From the definition of the transaction and the ACID feature that the implementation transaction must satisfy strong consistency, therefore, a theoretical —— BASE theorem that does not satisfy strong consistency is proposed. Ensuring the availability and high performance of storage management systems at the cost of strong consistency is the basic idea of BASE theorem. Where BASE represents: basic availability (Basically Available), soft state (Soft-state), and final consistency (Eventual Consistency) 。

(1) Basic usability: refers to the distributed system at the cost of losing part of the exception core function. For example, for a query system, the query request submitted by the user can be returned within 1 second, but when there is an exception, the correct result can still be returned within 2 seconds;

(2) Soft state: under the condition of ensuring the availability, the system can delay the consistent operation, which is the soft state;

(3) Final consistency: refers to the final consistency of the data in the system after a certain system delay;

In conclusion, BASE theorem guarantees usability on the basis of weakened consistency, and BASE theorem is a theory of anti-ACID.

1.2.3 Final consistency theory

Since the final consistency in the BASE theorem is more complex and more classified, it is listed as a separate theory. The final consistency can be given in the following five categories:

(1)Causal consistency (Causal consistency): As the name suggests, the causal relationship between operations should be consistent;

(2) Read-yourself-writes consistency (read the written consistency): that is, when reading the data, if you read the latest and updated data, you should also be able to read the last modified data;

(3) Session consistency (Session consistency): that is, after performing the data update

operation, the user can always read the latest data value in the same session;

(4) Monotonic read consistency (Monotonic read consistency): if a process reads the latest value from the system, then the system should not provide an old value for the process for any access request;

(5) Monotonic write consistency (monotonic writing consistency): refers to a system can ensure that the same process writing operation can be executed sequentially.

In conclusion, according to the CAP theorem, we know that perfect database system does not exist and focusing on strong consistency

And the availability of databases is unable to process big data efficiently. Therefore, No SQL databases focusing on availability and partition fault tolerance or consistency and partition fault tolerance emerge. Because No SQL database focuses on partition fault tolerance, No SQL database needs to be weakened in terms of consistency and usability. Therefore, the BASE theorem that makes trade-offs in terms of usability, the final consistency of consistency and the CAP theorem constitute the theoretical basis of No SQL database.

1.3 Key-value databases

The key value database is inspired by the data structure, which points to a data through Key and Value. Thus, the data access layer of the entire No SQL system only needs to store the keyprice. The biggest advantage of the key value model is that the data model is relatively simple, which only needs to query and modify the data through key. At present, the main key value databases are: Tokyo Cabinet / Tyrant, Redis, Voldemort, Oracle BDB, etc.

In the actual production environment, you need to decide how many Redis processes to open according to the actual demand. If the high concurrency requirements are higher, you may consider starting multiple processes on the same server. If the CPU resources

are relatively tight, just use a single process. The advantages of Redis are shown below

- Very high data reading and writing speed: the data reading speed can reach up to 110,000 times / s, and the data writing speed can reach up to 81,000 times / s.[1]
- Support for rich data types: support for key-value, Strings, Lists, Hashes, Sets, and Ordered Sets

And other data-type operations.

- Support data persistence: the data in memory can be saved in disk, and can be loaded again and used when restart.
- Atomic nature: Redis All operations are atomic nature.
- Support data backup: that is, data backup in master-slave mode.

Redis runs as a memory-based database, and caching is one of its most frequently applied scenarios. In addition, common Redis applications include obtaining the latest N data, leaderboard applications, counter applications, storage relationships, real-time analysis systems, and logging.

1.4 Document databases

The document database is the most similar to the traditional relational database in the No SQL database. Its main through JSON or class JSON format documents for storage. A document database can be considered an improved version of a key value database that allows nesting key values and documents to create indexes that a normal key value database cannot support. The current mainstream document databases include Couch DB and Mongo DB, Sequoia DB, and etc.

1.4.1 The difference between a document database and a key-value database

Similarities: both are key-value structures

Difference: in the key value database, we can only find the whole value through key, the database does not know what is the content in value, but translate and parse the value through the application; in the document database, we can not only find value through key, but also find the specific content through the corresponding key in document, value is transparent to the database.

For example, use two databases to store the same JSON file, in the key value database, we can only query the JSON file through customerid, but the specific content cannot be directly found from the database. Using the document database, we can directly query what the company is under the customerid.

1.5 Wide column database

Column storage database, that is, store data in column families. Column families, which combine multiple columns into one group, and most column storage databases support column families. The advantage of column families is that they are able to store similar columns together to improve their storage and query efficiency. At present, the main column storage databases are: Cassandra, HBase, Riak and etc.

A column-clustered database (Wide-Column Database) is a type of NoSQL database that uses tables, rows and columns for data storage (wide-column store). Unlike relational databases, the names and formats of columns may differ from rows to rows in the same table. Wide column stores can be regarded as a 2 D key value store.

Column-clustered databases are optimized for fast retrieved data columns and are commonly used for OLAP. Columned storage for database tables is an important part of analyzing query performance because it greatly reduces the overall disk I / O requirements and reduces the amount of data to be loaded from the disk. Another major feature of storing data by column is convenient to store structured and semi-structured data, and convenient to do data compression.

Wide column stores (Wide-column store) are different from earlier column stores

(Column-oriented store) because their two-tier structure does not use a column data layout. In real column storage, a column data layout is used so that each column is stored separately on disk. Wide-column storage usually supports the concept of column families stored separately. In a given column family, all data is stored in a line-by-line manner, such that the columns of a given row are stored together instead of each column individually.

1.6 Graph databases

The graphics library is used to store the relationship before the entity. The graph database consists of two elements: node and relationship. Each node represents an entity, such as a person, place, thing, or other data, and each relationship represents how the two nodes are associated. Because this structure can be used to model a variety of different scenarios, such as from road systems, device networks, the medical history of a population, or things defined by other relationships. The query graph is also called the "traversal" graph, and the more representative graph database is Neo4J.

1.7 The comparison between NoSQL databases and SQL databases

- Different usage scenarios: relational database is suitable for database query scenarios with particularly complex relationships, while non-relational databases is vice versa
- Support for the "transaction" feature: relational databases provide perfect support for transactions, but non-relational databases do not
- Query speed: The NoSQL database stores the data in the cache, and does not need to go through the SQL layer. The relational database stores the data in the hard disk, and the natural query speed is far less than the NoSQL database.
- Format of storage data: The storage format of NoSQL is key, value form, document

form, picture form, etc., so it can store various formats such as base types, objects or collections, while the database only supports the base type.

- Traditional relational databases such as My SQL guarantee strong consistency and usability at the expense of partition fault tolerance. The No SQL database, both Couch DB and Mongo DB, maintain the partition fault tolerance, which lays the foundation for the storage and processing of massive data.theoretical principle. This section will mainly explain these three theories.

2 When we should use SQL or NoSQL databases

2.1 Suitable scenarios for applying SQL databases

The application scenarios of relational database are mainly divided into two categories:
OLTP and OLAP

2.1.1 Online Transactions (OnLine Transaction Processing)

As the main application field of traditional relational database, OLTP is mainly used in daily transaction activities, such as transfer, deposit, withdrawal and other operations involved in bank counter operations.

characteristic

- Large throughput: a large number of short online transactions (insert, update, delete), very fast query processing.
- High concurrency, (quasi) real-time response.

Typical OLTP scenario

- Retail system
- Financial trading system
- Train ticket sales system
- Seckill activities

2.1.2 Online analytical processing (OnLine Analytical Processing)

The concept of OLAP online analysis and processing was first proposed by E.F.Codd in 1993 relative to the OLTP system. It refers to the query and analysis operation of data, usually the query and analysis of a large number of historical data. It involves a long history, a large amount of data, and the aggregation at different levels. aggregation operations make transaction processing operations more complex.

characteristic

- It mainly focuses on complex queries and answers some "strategic" questions.
- Data processing focuses on data aggregation, summary, group calculation, window calculation and other "analytical" data processing and operation. Use and analyze the data from multiple dimensions.

Typical OLAP scenario

- Reporting system, and the CRM system.
- Financial risk prediction and early warning system, anti-money laundering system.
- Data mart, data warehouse.

The report system is a platform or system that produces fixed periodic reports or reports fixed format report data, such as daily report, weekly report, monthly report and annual report, which provides electronic report data for business decisions.

CRM system, customer relationship management system [5] is a comprehensive business system platform, such as maintaining customers, storing customer related information, analyzing customer behavior, responding to customers, retention and market activity management.

Data markets are generally for the application needs of a department level requirements in an organization, such as the analysis requirements of the credit card department.

Data warehouse is an analysis platform system generated by the construction of the whole enterprise analysis and processing environment.

2.2 Suitable scenarios for applying NoSQL databases

NoSQL A generalized definition of non-relational data storage. It has broken the long-term unification of the relational database and the ACID theory. NoSQL Data storage does not require a fixed table structure, and there is usually no connection operation. In the big data access has a relational database of incomparable performance advantages. Although the definition of No SQL database is relatively vague and No SQL databases, but

No SQL Database has some common features, mainly including scalability, low cost and overhead, and high flexibility:

(1) scalability: traditional database is easy to longitudinal extension, namely through the extended server storage capacity to store huge amounts of data, however, longitudinal extension cost is high and performance bottleneck, and No SQL database can not only longitudinal extension, easy to lateral expansion at the same time, namely by increasing multiple servers to store huge amounts of data, the cost is low. Although the relational database can also be expanded horizontally by installing additional components, the installation of components brings about an increase in system overhead and reduced access efficiency;

(2) Cost: Compared with the commercial relational database, the No SQL database is mostly open source database, such as Mongo DB, Redis, etc., and the system use cost is obviously lower;

(3) Flexibility: the relational database needs to design data table structure before storing data, and the table structure cannot be modified after storing data; No SQL The database does not have fixed table structure and can change properties after data storing, with more flexibility;

(4) Availability: The No SQL database supports a large number of servers to build a cluster, a server in the cluster

After the down, you can use the internal mechanism of the system to pass related requests from the down server to other servers,

Enables the system to be accessed normally, with higher availability.

To sum up, compared with the relational database with strict data structure and convenient to store and manage structured data, No SQL database is more suitable for the management of three-dimensional public security monitoring with massive and unstructured characteristics of measuring the data.

2.3 Conclusion

In the face of the "three high" requirements mentioned above, relational databases have encountered insurmountable obstacles, and for web2.0 websites, many of the main features of relational databases are often useless, for example:

- Database transaction consistency requirements
- Real-time write and read real-time write requirements of the database
- Requirements for complex SQL queries, especially for multi-table association queries

Therefore, relational databases appear less appropriate in these increasing application scenarios, and non-relational databases emerge to solve such problems.

3 The two different approach to meet the need in Abertawe Entertainment company

3.1The design of relational database based on MySQL

From the task description, we can know that there are two entities required to implement

the relational database (MySQL), namely customer information and digital asset. In the customer information table, there are four fields in total. It is obvious that the other three fields are associated with the Name field and satisfy the atom, so we can set the Name as the main key. The same thing happens in the customer information table, and the other fields are associated with Title, so we can set Title as the main key. To indicate the interrelated relationship of digital asset and user, we can add the Name field to digital asset and use it as a foreign key. In terms of setting the check constraints and integrity constraints, considering the actual situation, stock amount cannot be empty and the default value is 0. Price values must be more than 0 and less than 10000.

Finally, we change the number of that digital asset by implementing the update operation for the stock amount, which means indirectly implementing the transaction operation.

3.2 The design of relational database based on MongoDB

Mongo DB Is a representative database in Nosql data. It is a document database and is currently free and open source. Due to the distributed storage, it has the advantages of convenient dynamic expansion and free storage mode. In the case of high load, the server performance can be guaranteed by adding more nodes, and Mongo DB

The storage mode is flexible, and the storage format is in the Json format, such as: {key1: value1, key2, value2} form, using the https protocol as the communication protocol.

Mongo DB Use the memory mapping file, and support the replica set. Mongo DB It has the following advantages:

(1) The load of writing data is higher. Compared with the security of database transactions, Mongo DB focuses more on increasing the efficiency of database transactions. If you need to quickly insert some data that do not require high safety requirements, such as the rainfall and water accumulation data here, then Mongo DB

will be more suitable as a database.

(2) For processing large-scale forms, the pressure on traditional databases is obvious. Especially when the size of the form exceeds 5 G, the table processing speed of MySQL forms will be greatly reduced

If the database is expanded through the MongoDB sharding mechanism, then the efficiency will be guaranteed.

(3) When used in an unstable and uncertain environment, the MongoDB can also operate relatively safely and reliably, thanks to the mechanism of the MongoDB copy set. By setting up the set of copies in different machines, one of them

MongoDB Downtime, other servers can automatically and quickly achieve failover.

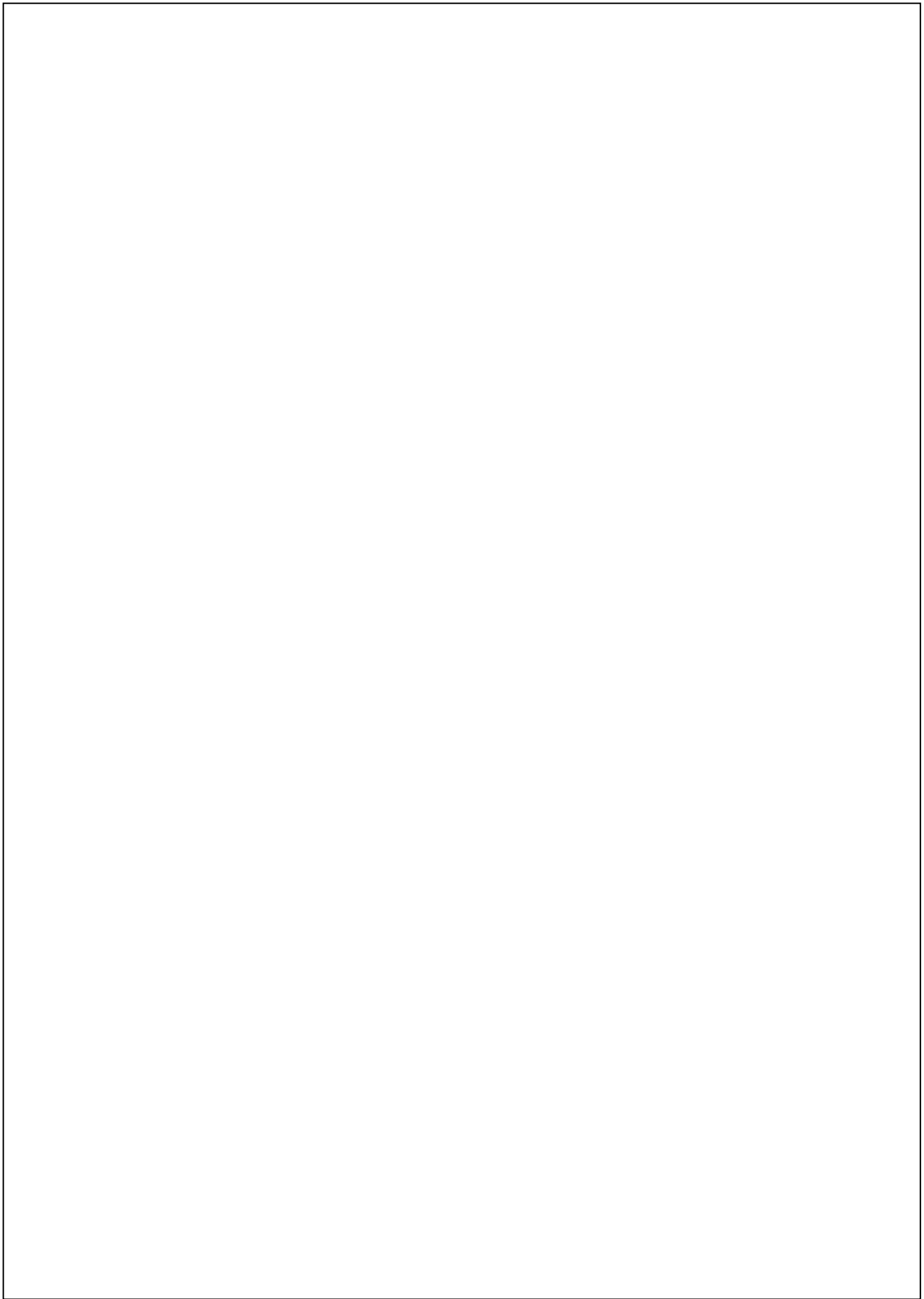
(4) When checking geographic information data, MongoDB query efficiency is faster. MongoDB Ability to index two-dimensional data. For example, the rainwater pipe and sewage pipe in this paper can establish an index, greatly improve the query efficiency, and can accurately obtain the location information of a certain pipeline.

(5) MongoDB supports a wide range of languages, and MongoDB can support the existing mainstream language platforms on the market, such as: C++, .NET, Python, Scala, Java, PHP, etc., and [23].

MongoDB Is a representative database in NoSQL data. It is a document database and is currently free and open source. Due to the distributed storage, it has the advantages of convenient dynamic expansion and free storage mode. In the case of high load, the server performance can be guaranteed by adding more nodes, MongoDB

The storage mode is flexible, and the storage format is in the JSON format, such as: {key1: value1, key2, value2} form, using the HTTP protocol as the communication protocol.

Different from MySQL, the data structure of MongoDB is relatively loose. We can convert SQL to export to file in DataGrip. Similarly, we can add update command for him to complete the transaction operation.



References

- [1]Matthew Tyson. How to CRUD with Node.js and MongoDB[J]. InfoWorld.com, 2021,
- [2]Kiran Fahd and Sitalakshmi Venkatraman and Fahd Khan Hammeed. A Comparative Study of NOSQL System Vulnerabilities with Big Data[J]. International Journal of Managing Information Technology, 2019, 11(4) : 1-19.

Data storage investigation

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

31 /100

Instructor

Generic Assessment Criteria	Marks available	Marks awarded
1. Engagement with Literature Skills Two reference articles is not sufficient for a Masters' level assignment. You must reference the claims that are being made in the report. I do not believe that OLAP will play a part in the solution to the case study scenario. You need more depth in your discussion and not more topic areas.	20	5
2. Knowledge and Understanding Skills Gaps in knowledge, with only superficial systematic understanding. Make use of diagrams when discussing the concepts behind the NoSQL and Relational database underpinnings. This will help considerably the reader understand the concepts	20	7

being explored. Also use the case study to take the general concepts and show you understand how to use them for a real world scenario. Consider using an Entity Relationship Diagram to illustrate the Relational Design and a defined schema for the NoSQL design.		
3. Cognitive and Intellectual Skills	15	5
Largely descriptive work, with superficial use of critical evaluation of research and methodologies. Absent or weak development of schemas for both Relational and NoSQL databases. The normalisation of data is not present for the relational database. You need far more than two tables in the relational database.		
4. Practical Application Skills	35	10
Rudimentary application of methods, materials, tools and/or techniques but without consideration and competence. You need to provide a relational database implementation with Functions, Views, Procedures, Function, Triggers, Tables, Constraints and Sequences. The MongoDB implementation is partial complete.		
5. Transferable Skills for Life and Professional Practice	10	4

Can communicate in a suitable medium but with some room for improvement. Use more references and support the text with appropriate citations. The report is rather general in nature. Make it specific to the case study. Have a title page at the top of the report, followed by a contents page with page numbers. Word can automatically create the contents page. The Abstract states <i>"This paper first introduces the contents and characteristics of four relational databases and compares them with relational databases."</i> It should compare four NoSQL database architectures with the Relational architecture.		

Assessment Mark (Assessment marks are subject to ratification at the Exam Board. These comments and marks are to give feedback on module work and are for guidance only until they are confirmed.)

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17
