A study of NoSQL Database for enterprises

Jeang-Kuo Chen[#*], Wei-Zhe Lee[#]

[#]*Dept. Information Management, Chaoyang University of Technology*
Taichung, Taiwan

[1]jkchen@cyut.edu.tw

[2]s10514613@cyut.edu.tw

Abstract— **Popularization of big data makes the enterprise need to store more and more data. The enterprise's database must access data as fast as possible, but RDB has the speed limitation due to the join operation. Many enterprises have changed to use NoSQL database which can meet the requirement of fast data access. However, there are more than hundreds of NoSQL databases. It is important to select a suitable NoSQL database for a certain enterprise because this decision will affect the performance of the enterprise operations. In this paper, 15 categories of NoSQL databases will be analysed to find out the characteristics of every category. Some principles are proposed to choose an appropriate NoSQL database for different types of enterprises.**

Keywords—**Big Data, NoSQL database, HBase, MongoDB, RDB**

I. INTRODUCTION

With the popularization of big data collection technologies and application fields, enterprises need to store more and more data than ever. The enterprise's database is desired to be accessed as fast as possible. However, RDB (relational database) suffers a bottleneck of speedup due to mass data join operations. Besides, in addition to the relational data storage format, other data storage formats such as key-value pairs, document-oriented, time series, etc. are proposed in many applications. Therefore, more and more enterprises have decided to use NoSQL databases to store big data [2][3][11].

However, there are more than 225 kinds of NoSQL databases [2]. How to choose an appropriate NoSQL database for a specific enterprise is very important because the change of database may affect the enterprise performance of the business operations.

This paper introduces basic concepts, compares the data formats and features, and lists the actual products for every category of NoSQL databases. In addition, this paper also proposes principles for different types of enterprises to choose an appropriate NoSQL database to solve the business problems and challenges.

II. RELATED WORK

NoSQL is the abbreviation of "Not Only SQL." That means if RDB is suitable to use then use it while if RDB is unsuitable to use, alternatives can be used [3]. The features of NoSQL databases are described as follows [2][3][11].

(1) Non-relational: NoSQL databases do not use Relational Database Model, neither support SQL Join operations.

(2) Distributed: data in NoSQL databases is usually stored in different servers and the locations of the stored data are managed by metadata.

(3) Horizontally scalable: the capacity of NoSQL database can be extended by increasing the number of servers.

(4) High data processing rate: The data processing rate of NoSQL database is higher than that of RDB.

According to the statistics of NoSQL databases official website [2], the current number of NoSQL databases has more than 225. Moreover, some NoSQL databases are widely used in many famous enterprises such as Google, Yahoo, Facebook, Twitter, Taobao, Amazon, and so on [3].

III. THE CATEGORIES OF NOSQL DATABASES

This section will explain the basic concepts of each category of NoSQL databases and analyse data formats or features that each category of NoSQL databases is suitable for processing.

A. Wide Column Store

This NoSQL database has a complex table schema described as follows [4][12][15][16].

(1) A *Row Key* is an identification that has a unique value used to identify a specific record, similar to the primary key of a relation in RDB.

(2) A *Timestamp* (abbreviated as ts) is an integer used to identify a specific version of a data value.

(3) At least one *Column Families* that have the format of "Family : Qualifier = Value," where "Family" is the name of a *Column Family*, "Qualifier" is the name of a *Column Qualifier*, and "Value" is a real value of a *Column Qualifier* stored in text.

An example is illustrated as follows. An inventory table of 3C products of *Wide Column Store* is shown in TABLE I, where

(1) Products_Inventory is the name of the inventory table which contains 2 Column Families, Products and Inventory, and has 2 records with the product codes P001 and P002 as the values of two *Row Keys*, respectively;

(2) an increasing integer ti (i=1..12) is the value of *Timestamp* for each *Column Qualifier* when a data value of a *Column Qualifier* is inserted into the table;

(3) *Column Family* Products includes four *Column Qualifiers*: classes, title, descriptions, price, and their data values, for example, are "TV," "LG 55 inch 4K LED TV," "TBD," "27000," respectively;

978-1-5386-7036-1/18/$31.00 ©2018 IEEE
DOI 10.1109/IS3C.2018.00116

IEEE computer society

(4) *Column Family* Inventory includes two *Column Qualifiers*: quantity, place, and their data values, for example, are "10," "1A," respectively.

According to the statistics of the DB-Engines Ranking Website [13], Apache Cassandra and Apache HBase are more widely discussed ones of *Wide Column Store* databases.

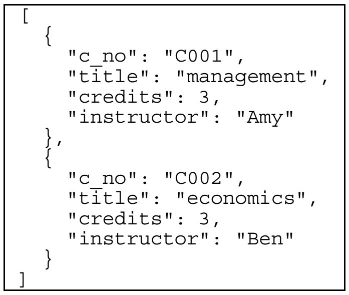
TABLE I
AN EXAMPLE OF A DATA TABLE IN WIDE COLUMN STORE.

Table Name: Products_Inventory

Row Key	ts	Column family Products	Column family Inventory
P001	t1	Products:classes = "TV"	
	t2	Products:title = "LG 55 inch 4K LED TV"	
	t3	Products:descriptions = "TBD"	
	t4	Products:price = "27000"	
	t5		Inventory:quantity = "10"
	t6		Inventory:place = "1A"
P002	t7	Products:classes = "Laptop"	
	t8	Products:title = "ASUS FX503VD i7 gaming laptop"	
	t9	Products:descriptions = "TBD"	
	t10	Products:price = "32000"	
	t11		Inventory:quantity = "20"
	t12		Inventory:place = "2A"

B. Document Store

This NoSQL database stores data with files of semi-structured documents which have specific formats such as XML (eXtensible Markup Language) or JSON (JavaScript Object Notation) [4][6]. An example of a *Document Store* table is shown in Fig. 1 which is a JSON file to store school curriculum data. There are two courses, management and economics, in this file. Each course contains four fields, c_no, title, credits, and instructor. According to the statistics of the DB-Engines Ranking Website [13], MongoDB and Couchbase are more widely discussed ones of *Document Store* databases.



```
[
  {
    "c_no": "C001",
    "title": "management",
    "credits": 3,
    "instructor": "Amy"
  },
  {
    "c_no": "C002",
    "title": "economics",
    "credits": 3,
    "instructor": "Ben"
  }
]
```

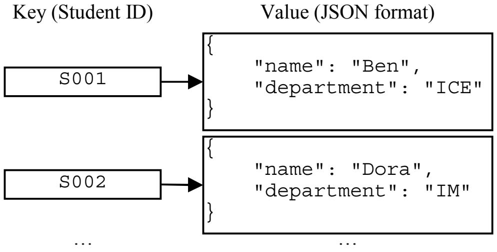
Fig. 1. An example of a data file in Document Store.

C. Key Value Store

The data in this NoSQL database is stored with the format of "Key → Value" [4], where

(1) "Key" is a string used to identify the unique "Value;"
(2) "Value" is the real data which can be number, string, or a JSON file, etc.;
(3) The user can search for a certain "Value" by a specific "Key."

An example of "Student ID Card rollcall System" is illustrated in Fig. 2. The system can sense a Student ID Card to identify the student owning the Student ID Card to be attending a class. A student ID is used as a "Key." The "Value" is the information of a student stored in a JSON format. According to the statistics of the DB-Engines Ranking Website [13], both Redis and DynamoDB are more widely discussed ones of *Key Value Store* databases.



```
{
    "name": "Ben",
    "department": "ICE"
}
```

```
{
    "name": "Dora",
    "department": "IM"
}
```

Fig. 2. An example of data stored in Key Value Store.

D. Graph Databases

This NoSQL database stores data based on a graphic structure. A European airline is illustrated as an example. The airline needs to store flight hours among some European nations. The data can be stored in a *Graph Database* as shown in Fig. 3. In this *Graph Database*, each vertex contains some data such as Nation, City and A2C_time (time from airport to city centre), and each edge represents the flight duration between two nations. According to the statistics of the DB-Engines Ranking Website [13], Neo4J and OrientDB are more widely discussed ones of *Graph Database* databases.

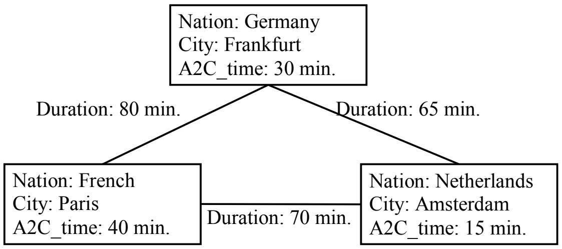


Fig. 3. An example of data stored in Graph Databases.

E. Multimodel Databases

The data format of this NoSQL database contains more than two data formats of the other categories of NoSQL databases [10]. According to the statistics of the DB-Engines Ranking Website [13], OrientDB and ArangoDB are more widely discussed ones of *Multimodel Databases* databases. OrientDB contains the data formats of *Object Database*, *Document Store*, *Graph Database*, and *Key Value Store*; while

ArangoDB contains the data formats of *Document Store*, *Graph Database*, and *Key Value Store* [2].

F. Object Databases

This NoSQL database combines the functions of object-oriented programming languages and traditional databases [1]. A web-based application system which provides users to order lunch boxes is illustrated as an example. The data in the *Object Databases* is described in the form of a class diagram as shown in Fig. 4 [20]. In Fig. 4, each rectangle is an object that includes both data items and data processing functions. For example, the object Customers has 4 data items (account, password, telephone, and e-mail) and two data processing functions (readData() and writeData()). According to the statistics of the DB-Engines Ranking Website [13], db4o and Versant are more widely discussed ones of *Object Databases* databases.

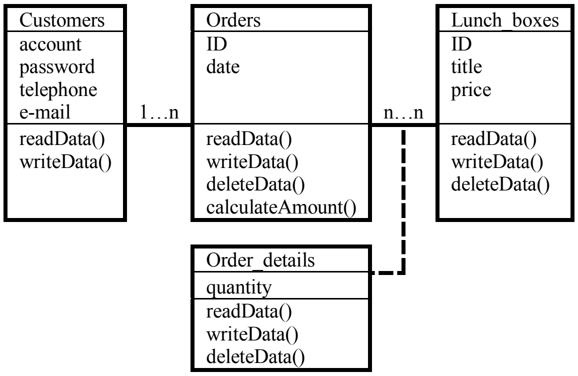


Fig. 4. An example of a Class Diagram in Object Databases.

G. Grid & Cloud Database Solutions

This NoSQL database stores recent access data in Random Access Memory (RAM) and uses Grid Computing to speed up the time of access data from a database [2]. According to the statistics of the DB-Engines Ranking Website [13], Hazelcast and Oracle Coherence are more widely discussed ones of *Grid & Cloud Database Solutions* databases.

H. XML Databases

The files stored in this category of NoSQL databases are based on the XML format [6]. An example of a school curriculum file stored in an *XML database* is shown in Fig. 5. In this XML file, there are two courses, AI and MMDB, which have course numbers (c_no), C001, C002, and credits, 3, 2, respectively. According to the statistics of the DB-Engines Ranking Website [13], Oracle Berkeley DB and BaseX are more widely discussed ones of *XML databases* databases.

```
<?xml version="1.0" encoding="UTF-8"?>
<Courses>
   <Course c_no="C001">
        <title>AI</title>
        <credits>3</credits>
        <remark>Teaching in English</remark>
   </Course>
   <Course c_no="C002">
        <title>MMDB</title>
        <credits>2</credits>
   </Course>
   <!-- ... -->
</Courses>
```

Fig. 5. An example of a data file in XML Databases.

I. Multidimensional Databases

The data in this NoSQL database is stored in a multidimensional array in order to analyse the value of each array element. Suppose a printing company stores data in the *Multidimensional Databases* as shown in Fig. 6 [18]. The printing company needs to analyse the total sales amount of printed products based on three dimensions: products, branches, and customer rank. For example, the company has two branches, Taipei and Taichung, two products, Copy Paper and Photo Paper, and two customer ranks, Platinum member and normal member. The boss of the printing company wants the total sales amount of each branch, each product, and each customer rank. According to the statistics of the DB-Engines Ranking Website [13], Intersystems Cache and GT.M are more widely discussed ones of *Multidimensional Databases* databases.

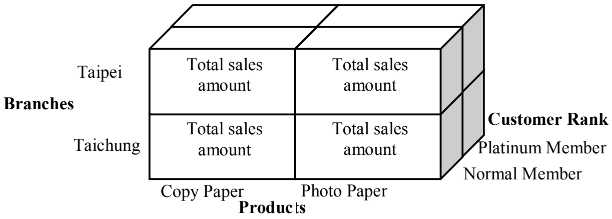


Fig. 6. An example of a three-dimensional array in Multidimensional Databases.

J. Multivalue Databases

This NoSQL database is suitable for storing data of multivalued attributes or composite attributes [19]. An example of student data is illustrated in a table of *Multivalue Databases* as shown in TABLE II. The schema of the table is Students(SID, Name, Society), where Name is a composite attribute composed of the two attributes, First_name and Last_name, Society is a multivalued attribute. There are three records in this data table, the name of each student is divided into two parts to save into the attributes, First_name and Last_name, respectively, and the attending societies of each student can have more than one values.

According to the statistics of the DB-Engines Ranking Website [13], jBASE and Model 204 Database are more widely discussed ones of *Multivalue Databases* databases.

TABLE II
AN EXAMPLE OF A DATA TABLE IN MULTIVALUE DATABASES.

Table Name: Students

SID	**Name**		**Society**
	First_name	**Last_name**	
S001	Cindy	Lin	{Pop music,Choir}
S002	Eric	Wu	{Choir, Poetry}
S003	Peggy	Lu	{Computer, Guitar}

K. *Event Sourcing*

This NoSQL database is suitable for storing events that occurred in the past in order to track the status of a specific event. An example about a lecture registration system to store the data in an *Event Sourcing* database is shown in TABLE III. In this table, the first two fields, Time and Person, can be considered as an event, and the last field Current Enrolment Number is used to track the number of people currently enrolled in the lecture [5]. According to the statistics of the DB-Engines Ranking Website [13], Event Store is the most widely discussed database of *Event Sourcing* databases.

TABLE III
AN EXAMPLE OF A DATA TABLE IN EVENT SOURCING [5].

Time	**Person**	**Current Enrolment Number**
2018/03/22 12:30	Amy	1
2018/03/25 10:40	Ruby	2
2018/03/28 13:20	Cindy	3

L. *Time Series Databases (TSDBs)*

This NoSQL database is designed to handle time series data [7][8]. An example of wave observation data is illustrated as follows. Assume that an observing station measures the wave height and wind direction once an hour and transmits the measurement result to a *TSDB*, and the results in 2017 are shown in TABLE IV [17]. According to the statistics of the DB-Engines Ranking Website [13], Informix Time Series Solution and influxdata are more widely discussed ones of *TSDBs* databases.

TABLE IV
AN EXAMPLE OF A DATA TABLE IN TSDBS [17].

Measurement Time	**Wave height (m)**	**Wind direction**
2017/01/01 00:00	0.6	East
2017/01/01 01:00	0.6	Southeast
…	…	…
2017/12/31 23:00	0.5	Northeast

M. *Scientific and Specialized DBs*

This category of NoSQL databases is designed to solve scientific and professional issues. For example, BayesDB allows users who have not been statistically trained to solve basic science problems, and GPUdb is a database suitable for distributed computing [2].

N. *Other NoSQL related databases*

The NoSQL databases in this category seem to be able to be categorized into several other categories mentioned earlier, but the official website of NoSQL database [2] categorizes them into this special category without giving any explanation for the characteristics of this category of NoSQL databases. Therefore, we have no way to know why this category is needed and the reasons why these NoSQL databases are assigned to this category. According to the statistics of the DB-Engines Ranking Website [13], eXtremeDB is the most widely discussed database in this category of NoSQL databases.

O. *Unresolved and uncategorized*

Any NoSQL database will be assigned to this category of NoSQL databases if it cannot be classified into any of the previously mentioned categories of NoSQL databases. According to the statistics of the DB-Engines Ranking Website [13], Adabas and CodernityDB are more widely discussed ones of *Unresolved and uncategorized* databases.

P. *Summary*

The basic concepts of each category of NoSQL databases have been described. Then, all the categories of NoSQL databases are analysed to get the results that each NoSQL database is suitable for processing certain features of data. The results are summarized in TABLE V.

TABLE V
SUMMARIZED RESULTS OF EACH CATEGORY OF NOSQL DATABASES

Categories of NoSQL Databases	**Suitable data features**
Wide Column Store	● Three-dimensional data. ● Applications that often search for specific field data.
Document Store	● Semi-structured files, such as XML, JSON, and so on.
Key Value Store	● One-dimensional data which is stored in Key-Value pairs.
Graph Databases	● Data stored in a graphic structure. ● Suitable for data of social network relations, recommendation systems, and so on.
Multimodel Databases	● Determine data features suitable processing based on the data format of a specific database.
Object Databases	● The object-oriented concepts are used to describe the data itself and the relationship among the data. ● Suitable for Computer Aided Design (CAD) and Office Automation.
Grid & Cloud Database Solutions	● Applications that needs to search recent access data frequently.
XML Databases	● Data stored in XML files.
Multidimensional Databases	● Applications that often analyze data in multiple dimensions.
Multivalue Databases	● Data with multivalued attributes or composite attributes.
Event Sourcing	● Data with events that occurred in the past for tracking the status of something.
Time Series Databases	● Data related to time series.
Other NoSQL related databases	● Unable to know.
Scientific and Specialized DBs	● Data suitable for scientific research or computing.
Unresolved and uncategorized	● Data based on the data format of a specific database.

IV. CHOOSE AN APPROPRIATE DATABASE

A. *The Principle of Database Selection*

If an enterprise prepares to choose a NoSQL database, it must understand the following questions according to the cultures and characteristics of the enterprise.

(1) Understand the current problems, goals, and challenges of the corporate operations database.

(2) Determine to continue to use current RDB or to change to use NoSQL databases according to business requirements and features of NoSQL databases.

(3) If changing to use NoSQL databases, select a suitable category of NoSQL databases based on the features and formats of the enterprise's operating data.

(4) When selecting a specific NoSQL database, we can first find out the NoSQL databases that are most frequently discussed on the Internet according to the statistics and evaluation of the DB-Engines Ranking website [13]. Finally, based on the advantages and disadvantages of these databases and enterprise's needs, we can select the most appropriate NoSQL database.

B. Database Selection Case 1

Suppose that a 3C shopping website uses an RDB to store data for a long period of time, and this RDB generates 300,000 records per day. Users reflect that the website is slower, and hope the data processing speed to be as fast as possible, so the business owner asks the information department staff to solve this problem.

The head of the information department traced the reason according to the boss's instructions, and found that the reason for the slower speed was not only the large amount of data generated every day, but also the main reason that many users need to merge (Join) several related tables with a large amount of data. Therefore, the supervisor suggested using the NoSQL database as a solution.

After the business owner agrees, the head of the information department will then decide which NoSQL database to use. The decision process is as follows:

(1) The most suitable category of NoSQL database is the Wide Column Store because access to the database often requires searching for data in a specific field.

(2) According to the DB-Engines Ranking Website [13], the Wide Column Store databases that are more commonly discussed on the Internet are Apache Cassandra and Apache HBase.

(3) According to the experimental results of Chen et al. [14], Apache HBase reads data less than Apache Cassandra. Therefore, Apache HBase is recommended as the NoSQL database used by the enterprise.

C. Database Selection Case 2

In order to understand the news reporting strategy of a peer, a famous newspaper must collect online news from various newspapers or media, so about tens of thousands of online news are stored for analysis every day. The information staff found that RDB could not provide quick access to a large amount of data in time; therefore, it is recommended to use the NoSQL database to solve the problem of too slow access rate.

In response to this question, the head of the information department must decide which NoSQL database to use for the newspaper company. The decision-making process is as follows:

(1) Because the newspaper needs to collect files generated by a large number of instant messages such as tens of thousands of online news and related readers' messages every day, it is necessary to replace the RDB with a NoSQL database.

(2) There are fifteen categories of NoSQL databases available, and the category found to be suitable for storing news multimedia materials is Document Store.

(3) According to the DB-Engines Ranking Website [13], the Document Store database that is often discussed on the Internet has two NoSQL databases, MongoDB and Couchbase Server. Because the former has a higher market share than the latter, it is recommended to use MongoDB as the NoSQL database for the company.

V. Conclusions

This paper introduces the basic concepts of each category of NoSQL database and analyses the characteristics of data that is suitable for each NoSQL database to process. This information can help specific companies to find the right database from hundreds of NoSQL databases when they abandon RDB and switch to NoSQL. Finally, two cases are used to illustrate how an enterprise chooses an appropriate NoSQL database.

References

[1] H. A. Chen, *Database System: Concept, Design, and Implementation*, 3rd ed., XBOOK MARKETING Co., Ltd., 2013.

[2] (2011) NoSQL databases [Online]. Available: http://nosql-database.org/

[3] S. J. Pi, Establish the cornerstone of Big Data: NoSQL Database technique, 2nd ed., TopTeam Information Co., Ltd., 2016.

[4] S. Dan, *NoSQL for Mere Mortals*, 1st ed., Pearson P T R, 2015.

[5] Microsoft Corp. (2018) Introducing to Event Sourcing [Online]. Available: https://msdn.microsoft.com/en-us/library/jj591559.aspx#sec1

[6] (2018) Document-oriented database (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/Document-oriented_database

[7] (2017) Time series database (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/Time_series_database

[8] (2018) Time series (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/Time_series

[9] (2018) Graph Databases (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/Graph_database

[10] (2017) Multi-model databases (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/Multi-model_database

[11] J. H. Lu, Challenge big data, how to process Big Data in Facebook, Google, Amazon? Use NoSQL to get 10 billion annual hard disk data, 2nd ed., TopTeam Information Co., Ltd., 2015.

[12] N. Dimiduk, and A. Khurana, *HBase in Action*, 1st ed., Oreilly & Associates Inc., 2012.

[13] SOLID IT Team (2018) DB-Engines Ranking [Online]. Available: https://db-engines.com/en/ranking

[14] C. Y. Chen, B. R. Chang, H. F. Tsai, and C. L. Guo, "Empirical Analysis of High Efficient Remote Cloud Data Center Backup Using HBase and Cassandra," in *Scientific Programming*, 2014, vol. 2015, article ID 294614, pp. 1-10.

[15] J. H. Lu, *Hadoop: Practical Technical Handbook*, 2nd ed., TopTeam Information Co., Ltd., 2014.

[16] L. George, *HBase: The Definitive Guide*, 1st ed., Oreilly & Associates Inc., 2011.

[17] (2018) Central Weather Bureau [Online]. Available: https://www.cwb.gov.tw/eng/index.htm

[18] (2016) Multidimensional Databases [Online]. Available: https://docs.oracle.com/cd/E12478_01/rpas/pdf/150/html/classic_client_user_guide/basic_rpas_concepts/multidimensional_databases.htm

[19] (2018) MultiValue (Wikipedia) [Online]. Available: https://en.wikipedia.org/wiki/MultiValue

[20] R. H. Wu, *Object-Oriented System Analysis and Design: An MDA Approach with UML*, 4th ed., BestWise Co., Ltd., 2013.