

# Research on Maiustream Web Database Development Technology

Yue Xu<sup>1, a</sup>

<sup>1</sup>Salesforce Service Cloud, Redmond, 98052, USA

<sup>a</sup>yxu.joyce@gmail.com

---

**Abstract:** With the rapid development of Internet technology, Web database occupies the core position in modern information system. This study provides an in-depth analysis of the current popular web database construction technologies, covering the application and unique technical attributes of traditional relational databases, new non relational databases, and cloud computing databases in web applications. At first, various types of web databases were systematically divided, and then focused on key aspects of database construction, including database architecture design, query efficiency improvement, and data security protection. Through a detailed analysis of current mainstream database technologies, the aim is to provide practical technical references for developers, help improve the operational efficiency of web systems, and ensure their security and stability.

**Keywords:** Web database; Relational database; Non relational databases; Cloud database; Database development.

---

## 1. Introduction

One of the core pillars of web applications is web database technology, which undertakes the important tasks of storing, managing, retrieving, and processing a large amount of information. In today's rapidly expanding information society, the level of development of database technology directly determines the operational efficiency of web applications and the interactive experience of users. With the increasingly diverse demand for databases, traditional relational databases, emerging non relational databases, and database technologies in cloud computing environments have emerged one after another, meeting the needs for data storage and processing in various application scenarios. This article aims to deeply analyze these mainstream web database technologies and interpret their key factors in the development process, in order to provide reference and guidance for the future field of web development.

## 2. Classification of Web Databases

### 2.1. Relational databases

The relational architecture in database systems relies on the relational model to store and regulate data in the form of rows and columns interwoven tables. The core advantage of this type of database is that it constructs clear data associations through foreign key associations between tables, and has excellent data consistency and integrity maintenance capabilities. Among numerous relational databases, MySQL, PostgreSQL, Oracle, and SQL Server are particularly well-known. It commonly uses Structured Query Language (SQL) for data operations and supports advanced transaction processing, multi table joint queries, elimination of data redundancy, and data recovery strategies. More importantly, the transaction management mechanism of relational databases ensures the atomicity, consistency, isolation, and persistence of transactions, thereby ensuring the stability and reliability of the entire system.

### 2.2. Non relational databases

Non relational databases refer to data storage systems that

break away from traditional row column database frameworks, typically using various forms of data organization such as key value mapping, document storage, wide column storage, or graph databases, exhibiting high flexibility, scalability, and excellent performance. This type of database aims to provide strong technical support for processing massive datasets, handling high-speed concurrent operations, and implementing distributed data storage. NoSQL databases tend to relax the strict restrictions of ACID transactions and instead place more emphasis on data access efficiency and system scalability, making them particularly suitable for dealing with data types that do not have fixed formats or have variable formats. Representative NoSQL databases include MongoDB, Redis, Cassandra, and Neo4j, which are widely used in many industries such as big data processing, real-time data processing, social media, and the Internet of Things. Thanks to its flexible architecture and decentralized nature, NoSQL databases can achieve rapid read and write of large-scale data, making them highly suitable for high-performance application requirements in distributed systems and cloud computing environments.

### 2.3. New type of database

With the continuous upgrading of information processing demands, in addition to common relational and non relational databases, a series of innovative database types have emerged one after another. These advanced database technologies have been deeply customized for various application scenarios, possessing the ability to handle more complex and diverse data structures. Graph databases focus on solving relationships between nodes and intricate data connections, while time series databases excel at quickly accessing and processing time related data sequences. Distributed ledger databases demonstrate unique advantages in decentralization and data immutability. These new database technologies are increasingly being applied in multiple industries such as massive data, the Internet of Things, and fintech, demonstrating their strong advantages in flexibility, efficiency, and scalability. Table 1 shows several types of new databases and their main characteristics.

**Table 1.** New Database and Its Characteristics

database system	data model	Typical application scenarios	characteristic
Neo4j	Graph database	Social networks, recommendation systems	Proficient in handling complex associated data with high query efficiency
InfluxDB	temporal database	Internet of Things, Monitoring System	Efficient storage and querying of time series data
BigchainDB	Blockchain database	Financial technology, supply chain	Support decentralization and data transparency, with tamper proof features
Snowflake	Cloud native data warehouse	Big data analysis and data warehousing	Support large-scale data parallel processing, suitable for cloud environments

Various new database technologies have emerged, achieving precise matching of data storage and management for diverse application scenarios, thus bringing richer options and higher mobility to the field of web development.

### 3. Core Elements of Web Database Development Technology

#### 3.1. Database Design and Modeling

In web database development, building an efficient and structurally optimized database is a crucial initial step aimed at creating a data system with high storage efficiency and reasonable structural layout based on business needs. The design process generally starts with requirement research, clarifying the types of data that the system needs to process and their inherent connections. Subsequently, developers use professional tools such as entity relationship diagrams (ER diagrams) to design the database architecture, transforming business rules into specific components such as database tables, columns, and primary keys, and establishing their interactive relationships. In database architecture design, the core point is the standardization of data processing, with the aim of reducing data redundancy and enhancing data accuracy and consistency.

During the development phase, engineers must carefully consider the complexity of system speed and data architecture. When building an online platform for a large number of users, it is necessary to carefully consider the scalability of the database, appropriate configuration of indexes, and sharding techniques. At the same time, it is necessary to strengthen data protection measures, improve transaction processing efficiency, and ensure that the system can adapt to future functional expansion during design.

The key stages of building a database architecture include analyzing business operation processes, mining user specific needs, and constructing data logic frameworks. Use entity relationship diagrams or Unified Modeling Language diagrams to depict the interrelationships between entities and establish a theoretical model for the database. Map theoretical models to tables, columns, and associations in a database, and maintain data integrity and consistency through normalization processing. Based on actual application requirements, select an appropriate database management platform and optimize and adjust data storage, index construction, and partitioning schemes.

#### 3.2. Database Query and Optimization

In the field of web database programming, a deep understanding and optimization of database queries is crucial,

as this step has a decisive impact on the running speed of applications and user experience. Faced with the surge in data volume and the increasing complexity of business logic, developers must address the challenge of how to achieve fast query execution and shorten response time. The core purpose of database query optimization is to improve query speed, reduce system resource consumption, and ensure that the database can still exhibit excellent performance when accessed concurrently by multiple users.

By building indexes, the efficiency of data search can be significantly accelerated, the database retrieval area can be effectively reduced, and the query speed can be improved. Common types of indexes include single field indexes, multi field indexes, and text indexes. Developers need to choose the appropriate indexing strategy based on specific application scenarios, while preventing uncontrolled index creation to avoid reducing data writing and updating performance. In addition to optimizing the index, optimizing the construction of SQL statements is also crucial. Reducing the use of subqueries, choosing appropriate association methods (such as inliers or left joins), and avoiding unnecessary function processing on index columns can significantly reduce the burden on the database.

One of the core components of a database system is the query optimizer, whose main function is to develop the optimal query plan and determine the specific execution process of SQL commands. Programmers can use built-in auxiliary tools in databases, such as the 'EXPLAIN' command in MySQL or the 'EXPLAIN Analyze' function in PostgreSQL, to deeply analyze query schemes and discover performance constraints, such as comprehensive scanning of tables or insufficient utilization of indexes. Meanwhile, the use of caching and database partitioning techniques are also key factors in improving query performance. Storing frequently accessed query results temporarily in memory can effectively reduce the number of redundant queries in the database. For tables with large amounts of data, partitioning technology can be used to divide the data into several logical units, which can narrow down the scope of queries and accelerate processing efficiency.

#### 3.3. Database Security and Management

The core of ensuring that web application data is not leaked, tampered with, and always available lies in the security protection and management of the database. The primary task is to implement access control, utilizing role-based access control mechanisms (RBAC) to finely allocate user permissions, ensuring that users can only access data resources that match their responsibilities. Furthermore, by

using data encryption methods to securely reinforce critical information, developers can apply encryption techniques such as comprehensive encryption and SSL/TLS encryption protocols during data storage and transmission to prevent data leakage or tampering. In addition, by logging and real-time monitoring of database operations, all database behaviors can be tracked, making it easy to quickly detect and handle any abnormal behavior, ensuring timely response to security incidents.

The planning of data backup and recovery is the core link to ensure database security. Implement regular automated data backup work and take strict measures to ensure the safe storage of backup materials, so that in case of data damage or loss, timely recovery can be carried out. In addition, database systems need to periodically install security updates and conduct vulnerability scans to prevent hackers from exploiting known vulnerabilities to launch attacks. Furthermore, by allocating resources reasonably and monitoring the operation status of the system, it is possible to effectively defend against distributed denial of service (DDoS) attacks, ensuring the stability and protection capabilities of the database in the face of high traffic.

## 4. Application of Mainstream Web Database Development Technologies

### 4.1. Development Technology of Relational Database

In the process of building web applications, relational database technology plays an indispensable role and is widely used in various fields such as online shopping, company management, banking, etc., all of which involve the processing of standardized data. The core of this technology lies in using table format to store information and relying on SQL language to perform operations such as adding, retrieving, modifying, and removing data. The main advantage of relational databases is that they can build relationships between data through foreign keys, ensuring effective protection and management of dataset integrity and consistency.

Using the order processing system of an online shopping platform as a case study, explain the specific application of relational database technology in practice. In this system, the order processing function involves managing consumer order information, product information, and payment progress data. Developers first build a relational data model, storing order data in the "Order Record Table" and product data in the "Product Information Table", while using the "Order Product Mapping Table" to establish the relationship between orders and products. With this table design, the system can efficiently retrieve order details, product catalogs, and related inventory status while maintaining logical relationships between data.

Once the user completes the purchase, the system will automatically capture order details, including key elements such as order number, customer unique identifier, and purchase time. The progress of the order will continue to change at various stages, such as pending payment, paid, and goods shipped. In order to ensure the integrity and unity of information, program developers have adopted database transaction processing technology. After the user completes the payment, the system must synchronously mark the order status as "paid" and reduce the inventory of related products accordingly. The transaction processing mechanism ensures

that these actions are either completed smoothly or rolled back in the event of any failure, thereby avoiding data mismatch issues caused by system operational errors.

In the pursuit of data retrieval speed, programmers often implement indexing strategies on frequently accessed fields such as order serial numbers and user IDs to accelerate the search process. Simultaneously streamlining SQL statements and avoiding tedious multi table interactions is also a key means to improve system response speed. Faced with high traffic challenges, relational databases, with their excellent transaction processing and data consistency guarantees, ensure the stability and security of data processing.

### 4.2. Non relational database development technology

Non relational database technology demonstrates its unique advantages in handling large-scale datasets, high concurrency requests, and unstructured information. Unlike traditional relational databases, non relational databases utilize a more flexible data storage architecture, such as key value storage, document storage, columnar storage, and graph storage, which makes them more efficient in handling massive amounts of data and adapting to changes in data structures. They are particularly suitable for real-time data processing, social media platforms, and the Internet of Things.

Taking a real-time user behavior log analysis system as an example, the system processes millions of user behavior records every day. Due to the limitations of traditional relational databases in table structure, developers have turned to document based non relational databases, where each log entry is stored as a separate document containing diverse fields such as timestamps, user IDs, and page access details. This storage strategy breaks free from the constraints of traditional table structures, making it more flexible to store log data in different formats.

In the field of data mining, calculating the daily browsing volume of each webpage is a key task. By adopting a non relational database and leveraging its decentralized network structure, log information can be efficiently processed and stored on numerous servers in a decentralized manner. Each server is responsible for managing a specific dataset. When performing query operations, hash algorithms are used to evenly distribute data to each server, achieving parallel query processing:

$$\text{hash}(\text{key})\%n$$

Among them,  $n$  is the number of nodes. Meanwhile, MapReduce technology can be used to aggregate and process log data. Assuming the number of visits to each page is  $x_i$ , the total number of visits is:

$$\text{TotalVisits} = \sum_{i=1}^n x_i$$

Among them,  $x_i$  is the access record of each page and  $n$  is the number of pages.

The system adopts a unique parallel processing mechanism and decentralized data storage architecture, which can maintain excellent performance even when processing massive amounts of data. With its flexible storage architecture and excellent scalability, non relational databases provide excellent data management strategies for contemporary web systems.

### 4.3. Cloud Database Technology

As a database service solution relying on cloud computing technology, cloud databases have been widely used in the field of modern web construction. It is based on a distributed system and utilizes a clustered management model to achieve flexible resource expansion, automatic data backup, continuous system availability, and billing based on usage, greatly enhancing the ability of web applications to handle large-scale concurrent requests and dynamic data changes.

In the order processing stage of e-commerce platforms, the system has to handle massive orders and payment requests every day, especially during promotional activities where the number of orders will increase sharply. Faced with such sudden access peaks, traditional database systems often struggle to maintain efficient operation. Cloud databases, with their automatic resource expansion capabilities, can adjust resource allocation in real-time based on changes in traffic. When order requests explode, the cloud database system will automatically increase computing and storage resources to ensure efficient application operation and system stability.

The elastic expansion of cloud databases can be represented by changes in processing power. As the number of order requests increases, the database automatically expands its processing capacity  $C_{cloud}$ . May vary according to actual needs:

$$C_{cloud} = f(R)$$

Among them,  $R$  represents the number of requests, and  $f$  represents the automatic expansion function of the cloud database. Cloud databases have the ability to adjust resources as needed and maintain the normal operation of the system in the event of a significant increase in request volume. Meanwhile, its automatic data backup and recovery mechanism is the core of ensuring data security. With the help of periodic automatic snapshot backups, the database can be quickly restored to its most recent normal operating state, effectively responding to system crashes or data loss situations. Backup data volume  $D_{backup}$ . Compared to the time period  $T_{backup}$ . The relationship between them can be expressed as:

$$D_{backup} = g(T_{backup})$$

Here,  $g$  it represents the setting of backup strategies in cloud databases. An appropriate backup plan can ensure the integrity and security of data. In the face of simultaneous access by multiple users, cloud databases utilize a decentralized architecture to distribute data across numerous nodes for storage and computation. This decentralized storage method enables the system to demonstrate excellent processing efficiency when dealing with a large number of concurrent requests. The overall processing capability of data  $Q_{total}$ . It is the processing capability of each node  $Q$ . The total sum:

$$Q_{total} = \sum_{i=1}^n Qi$$

This design pattern allows each node to divide the processing of specific data blocks, effectively improving the

parallel processing efficiency and retrieval speed of the system.

Relying on its flexible elastic expansion, decentralized structure, and automated data backup mechanism, cloud computing databases can cope with high loads, massive amounts of data, and constantly changing business needs, ensuring sustained efficiency and stability in performance. These prominent features make cloud computing databases an indispensable technology in modern web development, providing developers with stable and reliable support as well as highly flexible response strategies.

## 5. Epilogue

The rapid progress of Internet technology has promoted the core position of Web database technology in contemporary information systems. This study provides an in-depth analysis of relational, non relational, and cloud computing databases, elucidating their widespread applications and key characteristics in the field of web development. In terms of data integrity and transaction management, relational databases demonstrate their advantages and are mainly suitable for structured data processing; Non relational databases are an ideal choice for handling unstructured data and large concurrent requests due to their strong scalability and adaptability; Cloud databases rely on their flexible scalability and distributed structure to provide solid support for meeting changing data needs and massive business scales. In the future, the continuous innovation of database technology will greatly enhance the performance, security, and scalability of web applications, bringing more efficient tools and technical support to developers.

## References

- [1] Yang, Yi, Ying Xiong. "Simulation of Multi Database Parallel Scheduling Algorithm Based on Cloud Computing Platform." *Computer Simulation* 40.6 (2023): 459-462.
- [2] Tong, Chengquan, Yuxin Zhou. "Research on Cloud Database Selection in Complex Business Scenarios." *Telecommunications Science* 39.Z1 (2023): 39-46.
- [3] Hou, Benzong, Yongqiang Zhang, and Ying Shang. "Unstructured Data Extraction from Cloud Database Based on Natural Language." *Information Technology* 47.3 (2023): 57-63.
- [4] Chen, Fuhai, Kaiwen Fu. "Optimization Strategy for Cloud Database Architecture." *Mobile Information* 46.2 (2024): 129-131.
- [5] Han, Lu, Weiyu Chen, and Fei Zhang. "Non-relational Distributed Big Data Classification Method for Information Submission under Differentiated Needs." *Telecommunications Science* 39.6 (2023): 114-121.
- [6] Xu, Zhuxia, Weiwei Zhang, and Hao Liu. "Research on the Application and Expansion of Distributed Database in Meteorological Big Data Cloud Platform." *Meteorological, Hydrological, and Oceanic Instruments* 40.3 (2023): 1-4.
- [7] Wufu, Abuduani. "Analysis of Database Mining Technology Based on Web Database." *Computer Products and Distribution* 5 (2023): 41-43.
- [8] Liu, Yan, Li Zhang, and Lu Cao. "Bibliometric and Visual Analysis of Rimazolen Based on Web of Science Database." *Journal of Clinical Anesthesiology* 40.3 (2024): 282-289.