ISSN: 2834-4928

Research Article

Journal of Electrical Electronics Engineering

The Impact of Big Data on Database Management Systems

Micheal Ajayi*

Campbellsville University Louisville Kentucky, USA

***Corresponding Author**
Micheal Ajayi, Campbellsville university Louisville Kentucky, USA.

Submitted: 2025, Apr 09; **Accepted:** 2025, May 12; **Published:** 2025, May 27

Citation: Ajayi, M. (2025). The Impact of Big Data on Database Management Systems. *J Electrical Electron Eng, 4*(3), 01-14.

1. Introduction

Big data introduction has completely changed several industries by altering the way businesses gather, store, handle, and evaluate vast amounts of data. The foundation of data handling in every digital ecosystem, database management systems (DBMS), have been especially affected by this seismic change. The four Vs of big data: volume, variety, velocity, and veracity—all of which are increasing in volume—present both possibilities and difficulties for conventional DBMS solutions. This introduction explores the complex relationship between big data and database management systems (DBMS), emphasizing the development of data management technologies, the rise of new paradigms, and the commercial and societal ramifications.

Relational database management systems (RDBMS), which arrange data into tables with preset schemas, have historically been the mainstay of data administration. Because of its durability, support for Structured Query Language (SQL), and adherence to ACID (Atomicity, Consistency, Isolation, Durability) requirements, RDBMS solutions—such as Oracle, MySQL, and Microsoft SQL Server—have long been the industry standard. But the exponential expansion of data has also shown shortcomings in conventional RDBMS, especially in terms of real-time processing, scalability, and managing unstructured data. Because of its sheer size, big data requires more than small, gradual changes to standard databases. Rather, it has triggered a fundamental rethinking of database architecture, which has resulted in the creation of new data management solutions that are specifically tailored to effectively handle the intricacies of big data.

1.1 New Paradigms in Database Management Systems

NoSQL Databases: Because NoSQL databases provide more flexibility than standard RDBMS, its acceptance has been accelerated by the growth of large data, rather than only SQL databases. NoSQL databases are perfect for applications ranging from social media analytics to IoT data management because they can handle a broad range of data types, including semi-structured and unstructured data [1]. Examples of these databases include MongoDB, Cassandra, and Redis. NoSQL databases, in contrast to RDBMS, can grow horizontally, dividing data over several servers for more efficient management of big datasets.

spread Databases: Because large data is spread, systems such as Google Bigtable, Amazon DynamoDB, and Apache HBase have emerged as distributed database solutions [2]. These systems are designed to function over a large number of network nodes, offering fault tolerance and high availability [3]. Distributed databases use consistency models, replication, and data partitioning to provide reliable and easy access to data in large-scale settings.

NewSQL databases are designed to combine the finest features of NoSQL and RDBMS. They are intended to provide the scalability of NoSQL systems with the ACID qualities of conventional relational databases. VoltDB, Google Spanner, and CockroachDB are a few examples [4]. Applications that need strong transactional consistency and scalability are best suited for these systems.

Cloud-based Databases: Database management techniques have been greatly impacted by the scalability and flexibility of cloud computing [5]. Robust security features, automatic backups, and on-demand scalability are all provided by cloud-based databases like Microsoft Azure SQL Database, Amazon RDS, and Google Cloud Spanner [6]. They enable businesses to manage varying workloads without having to make costly on-premises infrastructure investments.

1.2 Problem Statement

Traditional database management systems (DBMS) are facing enormous issues as a result of the recent exponential expansion of data, also known as big data [7]. Managing enormous amounts of data, a variety of data kinds, quick rates of data production, and the need for real-time processing and analysis are some of these problems [8]. The inherent limits of traditional relational database management systems (RDBMS) in terms of scalability, flexibility, and performance when handling unstructured and semi-structured data typically leave them ill-suited to handle these needs.

Organizational resources, including both human and technological knowledge, are heavily taxed by the intricacy of effectively integrating, storing, and retrieving enormous databases [9]. Thus, there is a pressing need to investigate novel DBMS paradigms and technologies that can successfully tackle the difficulties presented by big data, guaranteeing reliable, scalable, and effective data management solutions.

1.3 Purpose of the Research

This study aims to examine how large data affects database management systems and assess how well-suited new DBMS technologies are to handle big data's accompanying issues. The purpose of this study is to:

Examine classic vs. Emerging DBMS: In terms of scalability, flexibility, performance, and capacity to manage a wide range of data types, compare and contrast emerging NoSQL, NewSQL, and cloud-based databases with classic RDBMS.

Analyze Scalability and Performance: When handling substantial amounts of structured, semi-structured, and unstructured data, evaluate the scalability and performance of different DBMS solutions [10]. This involves assessing these systems' capacity for managing dispersed data across several nodes and for horizontal scaling.

Analyze Real-time Processing Capabilities: Look at how well various DBMS systems can process data in real time, especially in terms of offering support for real-time analytics and decision-making.

Determine the social and commercial ramifications: Examine the real-world effects of using big data DBMS technology in the workplace, such as increased decision-making, operational effectiveness, and the creation of new business models [11]. Look at the social effects as well, such as advancements in science, healthcare, and urban development.

Challenges and Solutions: List the main difficulties in incorporating big data technologies—such as security, privacy, and compliance—into the current DBMS architecture. Make suggestions for possible fixes and effective strategies to get over these obstacles.

1.4 Relevance and Significance

The vital function that data plays in modern society makes this study relevant. Large amounts of heterogeneous data need to be managed and analyzed properly, as firms in all sectors grow more and more data-driven [12]. Even though they are fundamental, traditional database management systems (DBMS) are becoming less and less capable of managing the complexity of big data, which includes diverse data kinds, high velocity, and vast size [13]. A move toward more sophisticated and specialized DBMS technologies, such as NoSQL, NewSQL, and cloud-based solutions, is required due to this deficiency. For firms looking to stay competitive and maintain operational efficiency in a data-centric environment, they must comprehend the advantages and disadvantages of these technologies.

2. This Study has Several Significances:

• **Business Innovation and Efficiency:** By examining how big data affects DBMS, this study will assist companies in selecting the best data management options, which can boost decision-making, increase operational effectiveness, and spark the creation of new business models [14]. Big data management and analysis skills set companies up for success in terms of innovation, consumer personalization, and operational efficiency.

• **Technological Advancement:** By analyzing the most recent developments in DBMS made to manage large data, this study adds to the body of knowledge in the area of database technology [15]. The knowledge acquired from this research may direct database technology advancements in the future, guaranteeing that they satisfy the changing requirements of data management.

• **Benefits to Society:** Good big data management has a lot to do with society in addition to business. Through sophisticated analytics, better data management in the healthcare industry may result in better patient outcomes. Big data in urban planning may improve the quality of life by streamlining traffic and managing resources more effectively [16]. Large-scale data processing may speed up scientific study and lead to breakthroughs in a variety of sectors.

To put it briefly, this study is essential to the advancement of database management knowledge and practice, which will enable enterprises and society to fully use big data.

2.1 Research Problem

• How do traditional Relational Database Management Systems (RDBMS) compare with emerging NoSQL, NewSQL, and cloud-based databases in terms of scalability, flexibility, and performance?

• To what extent do NoSQL, NewSQL, and cloud-based DBMS solutions improve scalability and performance in managing large volumes of big data?

• How effectively do various DBMS technologies support real-time data processing and analytics?

• How do these technologies influence decision-making processes and the development of new business models?

2.2 Research Questions

These research questions are based on the comparison of traditional RDBMS with NoSQL, NewSQL, and cloud-based databases, and their impacts on scalability, flexibility, performance, real-time data processing, and business models:

• How do traditional RDBMS, NoSQL, NewSQL, and cloud-based databases differ in their approaches to scalability and performance?

• To what extent do NoSQL databases improve scalability and performance in managing large volumes of big data compared to traditional RDBMS?

• How do NewSQL databases balance the scalability and performance benefits of NoSQL with the consistency and transactional integrity of traditional RDBMS?

• What are the key flexibility differences between traditional RDBMS and modern database systems such as NoSQL, NewSQL, and cloud-based databases in handling diverse data types?

• How do cloud-based DBMS solutions leverage cloud infrastructure to enhance scalability, flexibility, and performance?

3. Barriers and Issues

To fully realize the promise of big data integration into database management systems (DBMS), enterprises must overcome several challenges and problems [17]. These issues affect the efficacy and efficiency of big data management on a technological, organizational, and regulatory level.

• **Scalability and Performance:** Ensuring that DBMS can scale effectively to manage the enormous quantities and high velocity of big data is one of the main issues (Yan & Ma, 2019). As data accumulates, traditional RDBMS often experience performance deterioration and problems with horizontal scalability. NoSQL and NewSQL databases provide answers, but setting them up and getting the most performance out of them still requires a lot of work and resources.

• **Data Variety and Integration:** Structured, semi-structured, and unstructured data are all included in the broad category of big data [18]. There are several obstacles to overcome to integrate these many data types into a coherent database system. It might be challenging to maintain data quality across several sources and ensure seamless integration of data.

• **Real-time Processing:** Another level of complexity is added by the need for real-time analytics and data processing [19]. Real-time data processing and analysis is a crucial feature for applications such as dynamic pricing, tailored marketing, and fraud detection that is often absent from conventional database management systems. It is quite difficult to implement systems that can process data in real time while preserving correctness and integrity.

• **Data Security and Privacy:** Maintaining strong data security and privacy is crucial given the growth in volume and sensitivity of data. Organizations have to deal with a complicated web of regulations, such as the CCPA and GDPR, which demand strict data protection practices [20]. Vigilant data governance procedures and advanced security policies are needed to ensure compliance while handling big databases.

4. Literature Review

Seref SAGIROGLU and Duygu SINANC (2013) Reviewed A Paper on Big Data

Description of the Research

Big data is a term for massive data sets having large, more varied, and complex structures with the difficulties of storing, analyzing, and visualizing for further processes or results. The process of researching massive amounts of data to reveal hidden patterns and secret correlations is called big data analytics [1]. Big data and its analysis are at the center of modern science and business. These data are generated from online transactions, emails, videos, audio, images, click streams, logs, posts, search queries, health records, social networking interactions, science data, sensors, and mobile phones and their applications. They are stored in databases that grow massively and become difficult to capture, form, store, manage, share, analyze, and visualize via typical database software tools [1].

In 2012, The Human Face of Big Data was accomplished as a global project, which centered on real-time collection, visualization, and analysis of large amounts of data. According to this media project, many statistics are derived. Facebook has 955 million monthly active accounts using 70 languages, and 140 billion photos. Uploaded, 125 billion friend connections, every day 30 billion pieces of content and 2.7 billion likes and comments have been posted. Every minute, 48 hours of video are uploaded and every day, 4 billion views are performed on YouTube. Google supports many services as both monitor 7.2 billion pages per day and processes 20 petabytes (1015 bytes) of data daily also translating into 66 languages [3].

Big Data requires a revolutionary step forward from traditional data analysis, characterized by its three main components: variety, velocity, and volume. Variety makes big data big. Big data comes from a great variety of sources and generally has three types: structured, semi-structured, and unstructured. Structured data inserts a data warehouse already tagged and easily sorted but unstructured data is random and difficult to analyze [4]. Semi-structured data does not conform to fixed fields but contains tags to separate data elements. Volume or the size of data now is larger than terabytes and petabytes. The grand scale and rise of data outstrips traditional storage and analysis techniques. Velocity is required not only for big data but for also all processes. For time-limited processes, big data should be used as it streams into the organization to maximize its value [4]. Examples of big data are available in our astronomy, atmospheric science, genomics, biogeochemical, biological science and research, life sciences, medical records, scientific research, government, natural disaster and resource management, private sector, military surveillance, private sector, financial services, retail, social networks, weblogs, text, document, photography, audio, video, click streams, search indexing, call detail records, POS information, RFID, mobile phones, sensor networks, and telecommunications. Organizations in any industry that have big data can benefit from its careful analysis to gain insights and depths to solve real problems [2].

4.1 Research Method

Most enterprises are facing new data, which arrives in many different forms. Big data has the potential to provide insights that can transform every business. Big data has generated a whole new industry of supporting architectures such as MapReduce. MapReduce is a programming framework for distributed computing that was created by Google using the divide and conquer method to break down complex big data problems into small units of work and process them in parallel [5].

Hadoop was created inspired by BigTable which is Google's data storage system, Google File System, and MapReduce. Hadoop is a Java-based framework and heterogeneous open-source platform. It is not a replacement for database, warehouse, or ETL (Extract, Transform, Load) strategy. Hadoop includes a distributed file system, analytics and data storage platforms, and a layer that manages parallel computation, workflow, and configuration administration. It is not designed for real-time complex event processing like streams [6]. HDFS (Hadoop Distributed File System) runs across the nodes in a Hadoop cluster and connects the file systems on many input and output data nodes to make them into one big file system [7].

HPCC (High-Performance Computing Cluster) Systems distribute data-intensive open-source computing platforms and provide big data workflow management services. Unlike Hadoop, HPCC's data model is defined by the user. The key to complex problems can be stated easily with a high-level ECL basis. HPCC ensures that ECL is executed at the maximum elapsed time and nodes are processed in parallel. Furthermore, the HPCC Platform does not require third-party tools like Greenplum, Cassandra, RDBMS, Oozie, etc [2].

4.2 Findings

The amount of data has been increasing and data set analyzing become more competitive. The challenge is to not only collect and manage vast volumes and different types of data but also to extract meaningful value from it. Also needed, are managers and analysts with an excellent insight of how big data can be applied. Companies must accelerate employment programs while making significant investments in the education and training of key personnel [1]. Through the TDWI Big Data Analytics survey, benefits of big data are better-aimed marketing, more straight business insights, client-based segmentation, recognition of sales and market chances, automated decision making, definitions of customer behaviors, greater return on investments, quantification of risks and market trending, comprehension of business alteration, better planning, and forecasting, identification consumer behavior from clickstreams and production yield extension. Big Data Analytics survey, there are several challenges for big data: data growth, data infrastructure, data governance/policy, data integration, data velocity, data variety, data compliance/regulation, and data visualization [7].

4.3 Conclusion

An overview of big data's content, scope, samples, methods, advantages, challenges, and discusses privacy concerns have been reviewed. The results have shown that even if available data, tools, and techniques are available in the literature, there are many points to be considered, discussed, improved, developed, analyzed, etc. Besides, the critical issue of privacy and security of big data is the big issue will be discussed more in the future.

Manar Abourezq *et al*., (2016) presented an overview of Database-as-a-Service for Big

5. Data

Description of the Research

Organizations are gathering more and more information, for various purposes: analysis to ameliorate their market position and offer better services to their customers, fraud detection, scientific projects like in genomics, legal reasons (for example, Moroccan firms are required by law to store ten years of financial data), etc [8]. This flow of data, which has been referred to as a flood or a tsunami, can't be managed using a classical approach and has led to the emergence of a new buzzword: Big Data. Almost all major IT leaders invested in various Big Data projects, from Google's BigQuery and Datastore to Amazon's Elastic MapReduce, to Facebook's Cassandra, Yahoo!'s PNUTS, etc [9].

Cloud computing has a leverage effect on Big Data, providing the computing and storage resources necessary for Big Data applications. The inherent characteristics of Cloud Computing, such as elasticity, scalability, automation, fault tolerance, and ubiquity offer an ideal environment for the development of Big Data applications [10]. Cloud Computing is an established computing paradigm that gained importance in the last decade. It refers to the utilization of storage and computation resources as a utility. There is a great tendency to opt for using IT as a service. It is estimated that more than 80% of Internet users use Cloud Computing in one form or another, from email services to different business applications as a service, all through data storage, development platforms, etc [12]. This usage percentage is even greater when it comes to companies: In a survey conducted by RightScale in January 2015, 93% of respondent companies confirmed using Cloud Computing, which shows that the latter is steadily advancing to become an integral part of companies and individuals use of IT [12]. Although the emergence of Cloud Computing is relatively new, the idea of delivering computing as a utility dates back to as far as the 1960s, when pioneers like John McCarthy, Leonard Kleinrock, and Douglas Parkhill predicted that, just like water, electricity, or the telephone, computing resources will someday be used as a public utility.

Cloud Computing's major deployment models are public, private, community, and hybrid resources since all resources are exclusively intended for the sole use of the organization. It also allows organizations to manage the security aspect of the cloud [13]. A Community Cloud is a private Cloud that is shared by organizations belonging to the same community, for example, many departments belonging to the same University, or many companies that want to use a specific application that the provider is going to offer solely to them. A Hybrid Cloud is composed of two or more of the Cloud models previously presented, interconnected by standard or proprietary technologies. As for service models, the major ones are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [16]. A Public Cloud is a deployment model in which cloud services are provided via a public network, usually the Internet. Examples include Amazon's Elastic Compute Cloud (EC2), Google's App Engine, and Microsoft's Azure. A Private Cloud is provided for the sole use of an organization that can either choose to be responsible for managing it or delegate its management to a third party. The organization can also choose to host it on-premise or off-premise. A variation of this deployment model is the On-Site Private Cloud, where the cloud is hosted and managed by the organization to which it is destined [19]. The main advantage of both models is that there are no restrictions on bandwidth IaaS provides basic virtualized resources, namely networking (network connections, bandwidth, IP addresses), virtual servers, and virtual storage space. This infrastructure will be completed by clients with the various blocks necessary and used to run their applications. The provider manages the underlying infrastructure, while it is up to the user to handle anything other than the hardware part of the architecture. Although IaaS management is majorly incumbent to users, it is the model that satisfies best interoperability and portability needs, since users can compose the various blocks of the infrastructure used.

PaaS is built on top of IaaS by adding a software layer to offer a development environment that can be used by clients to build and deploy their applications. It provides various development tools, such as APIs, for users to develop their applications. Clients can control the deployment and hosting environment of their applications without having to manage the underlying infrastructure [15]. Prominent PaaS include Salesforce's Force.com, Google App Engine, and Microsoft Azure. SaaS is arguably the most known and used cloud service model. It offers remote access to applications running in the Cloud, through various devices. Users seamlessly access ―ready-to-go‖ applications without needing to invest or manage the underlying infrastructure, buy software licenses, handle updates and patches, etc. The provider is responsible for the smooth running of the applications and the maintenance of the underlying infrastructure. Prominent SaaS include Google Drive and Salesforce CRM [15].

5.1 Research Method

An ever-growing number of companies found themselves swamped with the large amount of data generated and stored for different purposes (user-based preference suggestions, business analysis...). Storing and retrieving data becomes a costly and complex operation, involving investments in infrastructure and database managers [11]. It is only normal then that the question of outsourcing data was one of the earliest to surface with the emergence of Cloud Computing, which led to the DataBase as a Service (DBaaS) model. DBaaS can be simply defined as ―a paradigm for data management in which a third-party service provider hosts a database and provides the associated software and hardware support [53]. Companies using this model outsource all database management operations, from installation to backups, to the provider, and focus on developing applications [12]. They can access their database instances on-demand, using querying interfaces or programming tools. The increasing use of Cloud Computing, and especially SaaS, called for rethinking the persistency layer [16]. The inherent characteristics of cloud computing, such as elasticity, scalability, self-service, and easy management make traditional RDBMS not fully adapted for applications that run-in cloud environments. Early solutions tried extending existing DBMS to support high scalability, but it only led to complex solutions with poor performance [54]. Leader IT operators, such as Google, Yahoo!, and Facebook, chose to implement their data management solutions, namely Bigtable, PNUTS, and Cassandra. Various other databases provided as DBaaS were developed from scratch to integrate the advantages of the cloud, except a few providers who offer established relational or NoSQL databases, such as MySQL, PostgreSQL, MongoDB, and Redis, as a service. Database as a Service (DBaaS) is one of the Cloud Computing models that is most suitable for Big Data [18]. In this model, it is possible to use a database as a service and benefit from the high scalability and storage capacity offered by the Cloud, without having to install, maintain, upgrade, backup, or manage the database or the underlying infrastructure. DBaaS is a different concept from the concept of cloud databases, which is beyond the scope of our paper [19]. In this concept, users can either upload their machine image, with the database installed, to the cloud infrastructure or use a ready one offered by the provider. In both scenarios, the various database management operations are incumbent on users. Datawarehouse Cloud solutions are also beyond the scope of this paper.

The most prominent databases that are DBaaS are Cloud Bigtable and Cloud Datastore: Cloud Bigtable is a DBaaS based on Bigtable, a highly scalable, distributed, structured, and highly available column database developed by Google that has been used internally since 2003 to store the data of numerous Google projects (Google Finance, Google Analytics, Google Earth, etc.). Bigtable was made publically available as Cloud Bigtable in May 2015 [43]. A row key is an arbitrary string and is the unit of transactional consistency in Bigtable. Rows with consecutive keys are grouped into tablets, which are the units of distribution and load balancing [17]. A column key is also an arbitrary string, and column keys are grouped into column families, the unit of access control. Timestamps are used to manage data versioning. A cell can store different versions of the same data, each referenced by a timestamp. Older data is garbage-collected depending on the user's specifications. Bigtable relies on Google File System (GFS), a scalable distributed file system, for storing data in SSTable file format. An SSTable is a file of key/value string pairs that is sorted by keys. It is used to map keys to values. Bigtable also uses Chubby, a highly available and persistent distributed lock service, for synchronizing data access [56]. A Chubby service has four replicas and one master replica [13].

Cloud Datastore is a NoSQL, schemaless, highly scalable, and highly reliable database for storing non-relational data developed by Google as a part of the App Engine. The main motivation for its development is to answer the need for high scalability that couldn't be met by traditional relational databases [18]. It supports basic SQL functionalities, including filtering and sorting. Other functionalities like table joins, sub-queries, and flexible filtering are not supported. Cloud Datastore is based on another Google solution, namely Megastore, which is built on Bigtable. Megastore ensures strong consistency. It replicates data across multiple geographically distributed data centers using an algorithm based on a distributed consensus algorithm, Paxos. for committing distributed transactions. It also implements a two-phase commit (2PC) [60] for committing atomic updates. Unlike 2PC, Paxos doesn't require a master node for committing transactions [20]. Instead, it ensures that only one of the proposed values is chosen and, when it is, that all the nodes forming the cluster get the value. Thus, all future read and/or write access to the value will give the same result. For each new transaction, Megastore identifies the last transaction committed and the responsible node then uses Paxos to get a consensus on appending the transaction to the commit log. Megastore is built on Bigtable to overcome the difficulty of use in applications that have relational schemas, or that need to implement strong consistency. An amelioration to Megastore is Spanner, a highly scalable, globally distributed, semi-relational database where queries are done in an SQL-like language and offers better write throughput [15]. Though Spanner is not offered as a service to developers, it is used internally by Google as the backend of F1, Google's distributed RDBMS supporting its online ad business. However, there is a project for building an open-source version of Spanner, CockroachDB [2].

5.2 Findings

There are various databases offered as a service by many Cloud providers. This model of use, namely DBaaS, offers many advantages both to users and providers. Users find themselves exempt from upfront investments and relieved from the burden of installing, running, and administrating their databases. As for providers, the costs of providing their service are optimized, especially in the case of multi-tenancy. The concern for security and privacy in cloud environments is enhanced by the large volume of datasets managed by Big Data [10]. And just like DBaaS removes the burden of database installation and management, it also ensures the security of data. DBaaS providers implement different levels of security, starting from identity and access management, to data encryption, all through assuring the physical security and monitoring of data centers. In addition to securing data while being stored in data centers, it is crucial to ensure its transfer to and from client applications, which can be implemented using cryptographic protocols like TLS or SSL. Providers like Amazon, Google, Microsoft, IBM, and Rackspace have achieved the ISO/IEC 27001 certification for their cloud platforms [9].

5.3 Conclusion

Big Data has emerged as one of the most important technological trends for the current decade. It challenges the traditional approach to computing, especially regarding data storage. Traditional clustered relational database environments prove to be complex to scale and distribute to adapt to Big Data applications and new solutions are continually being developed. One of the most adapted answers to Big Data storage requirements is Cloud Computing, and more specifically Database as a Service, which allows storing and managing a tremendous volume of variable data seamlessly, without the need to make large investments in infrastructure, platform, software, and human resources. In this context, our article presents a benchmark of the main database solutions that are offered by providers as DataBase as a Service (DBaaS) [15]. Cloud Computing and Big Data are entwined, with Big Data relying on Cloud Computing's computational and storage resources, and Cloud Computing pushing the limits of these resources. New extensions of Cloud Computing are emerging to further enhance Big Data, especially Fog Computing and Bare-Metal Cloud. Fog Computing uses edge devices and end devices, such as routers, switches, and access points to host services, which minimizes latency. This proximity to end users, along with its wide geographical distribution and support for mobility makes Fog Computing ideal for Big Data and the Internet of Things applications [10].

Mardianil et al., (2023) presented a review on The Impact of Scalability and Consistency Management on Database Management System Performance in a Big Data Environment in Indonesia

6. Description of the Research

The efficiency of data management systems, including traditional DBMSs and contemporary BDEs, directly affects an organization's ability to leverage insights, streamline processes, and dynamically respond to market changes. These advancements have created a need for organizations to optimize their enterprise landscape by implementing modern technology solutions, such as Edge Computing Platforms, which enable real-time processing of massive amounts of data at the edge of devices [22]. Extracting actionable insights from data promptly empowers enterprises to be more efficient and effective in their operations. The interplay between data, technology, and business transformation is crucial for organizations to address their top challenges and create value for both the enterprise and its customers. Managers in the field of business informatics must understand and actively participate in these digital innovations to effectively direct strategies in the changed business environment. Scalability and consistency management in tech start-ups, particularly in the Indonesian landscape, is an area that lacks specific implications in the literature. Startups face challenges in scaling operations and maintaining data integrity, making the delicate balance between scalability and consistency management crucial for their continued success [24]. However, there is a noticeable gap in the literature regarding the implications of scalability and consistency management in tech start-ups, especially in the Indonesian context. Scalability in database management systems (DBMS) refers to the system's ability to handle increasing volumes of data, users, and transactions without compromising performance [26]. There are two primary scalability models: vertical scaling, which involves increasing the capacity of a single server, and horizontal scaling, which focuses on distributing the workload across multiple servers. Studies have explored the implications of scalability on different types of databases, including traditional relational databases and modern NoSQL databases. Adopting horizontal scaling in cloud-based relational databases has been found to significantly improve system performance and response times, particularly under high user loads [29].

6.1 Research Method

Data was collected through primary methods to ensure a comprehensive understanding of Indonesia's startup ecosystem. The methods used included: Distributed to IT professionals, database administrators, and decision-makers within the sampled startup companies. This approach guarantees a representative picture of Indonesia's startup scene [25]. The survey was designed to collect information on scalability measures, consistency management strategies, and perceptions of overall system performance. Likert-scale questions were included to capture quantitative data on these aspects [23].

The collected data underwent rigorous analysis using Structural Equation Modeling with Partial Least Squares (SEM-PLS) [24]. This statistical technique was chosen for its suitability in analyzing complex relationships between latent and observed variables, making it appropriate for assessing the diverse interdependencies within the startup ecosystem. The measurement model assessed the reliability and validity of the constructs, confirming that the chosen variables accurately represent the underlying concepts of scalability, consistency management, and system performance perceptions [25]. The use of SEM-PLS allowed for a detailed examination of the data, providing insights into the structural relationships within Indonesia's startup scene and identifying key factors influencing scalability and performance in various technological and industrial contexts.

6.2 Findings

The collected data underwent a thorough analysis using Structural Equation Modeling with Partial Least Squares (SEM-PLS). This approach was selected for its ability to handle complex relationships between latent and observed variables, making it well-suited for assessing the interdependencies within the startup ecosystem [30]. The loading factors demonstrate strong Mapanga correlations between the latent variables and their observable indicators. The Variance Inflation Factor (VIF) values are within acceptable boundaries, indicating no multicollinearity issues among the variables. High Cronbach's Alpha and Composite Reliability indicate robust internal consistency.

The model also exhibits a good Average Variance Extracted (AVE), confirming convergent validity. Demonstrates convergent validity and high internal consistency. The measurement approach is reliable for assessing consistency management within startups. The Database Management System Measurement Model Shows excellent internal consistency, with high Cronbach's Alpha and Composite Reliability [22]. The model also indicates good convergent validity. Big Data Environment Measurement Model Exhibits outstanding internal consistency and convergent validity. The measurement model is reliable for assessing the impact of big data environments on startups. Interrelations and Multicollinearity show the VIF values highlight the interactions between major factors in the study while there is considerable multicollinearity between Scalability and Consistency Management with the Database Management System, the overall VIF values remain within an acceptable range [31]. This indicates a minimal correlation between internal relationships, allowing for a confident assessment of the effects of Scalability, Database Management System, Consistency Management, and Big Data Environment independently.

6.3 Conclusion

The measurement models for Scalability, Consistency Management, Database Management System, and Big Data Environment show strong reliability and validity. The absence of significant multicollinearity issues ensures that the models can be used to independently assess the effects of these factors on the startup ecosystem in Indonesia. The rigorous analysis confirms the robustness of the models and provides valuable insights into the key determinants of scalability and performance in various technological and industrial contexts.

Mapanga et al., (2013): Database Management Systems: A NoSQL Analysis

7. Description of the Research

In the 1970s cord came up with the relational theory that led to the development of the relational Database Management Systems (RDBMS) as a solution to the challenges posed by the flat file database system in the earlier years. Storage of data in RDBMS was done using Tables. Standard fields and records are represented as columns (fields) and rows (records) in a table. Their major advantage was the ability to relate and index information [32]. Security was enhanced in RDBMS and they were also able to adapt to considerable growth of data. Structured Query Language, SQL is the programming language used for querying and updating relational databases. For a long time, RDBMS has been the preferred technique for data management purposes. However, RDBMS's inability to handle modern workloads has given rise to scalability, performance, and availability problems with its rigid schema design [33]. Businesses all over the world, including Amazon, Facebook, Twitter, and Google have adopted new ways to store and scale large amounts of data hence the move away from the complexity of SQL-based servers to NoSQL database Systems. NoSQL is a class of database management systems that have been designed to cater to situations in which RDBMSs fall short. It is different from the traditional relational databases mainly in that it is schema-less [34]. This makes it suitable to be used for unstructured data. These engines usually provide a query language that provides a subset of what SQL can do, plus some additional features. Among several capabilities of NoSQL databases are managing large streams of non-relational and unstructured data, fast data access speeds, and availability of data even when the system is operating in degraded mode due to network partitions. NoSQL databases provide near-endless scalability and great performance for data-intensive use cases. However, with so many different options around, choosing the right NoSQL database for your interactive Web application can be tricky [33].

7.1 Research Method

The NoSQL database approach is characterized by flexibility in the storage and manipulation of data, performance improvements, and allowing for easier scalability. Many different types of these NoSQL databases exist, each one suited for different purposes. Examples include MongoDB whose deployments are at Foursquare, Disney, bit.ly, Sourceforge, CERN, The New York Times, and others. Hadoop (Apache), Cassandra was primarily used by Facebook for their Inbox Search [35]. Afterward, it was open-sourced and now it is an Apache Software Foundation top-level project, being used by Digg, Twitter, Reddit, Rackspace, Cloudkick, Cisco, and others. DynamoDB is used by Amazon, Voldemort is used by Amazon, and Neo4J is used by Adobe Cisco, etc. While RDBMS is transaction-oriented and based on the ACID principle, NoSQL makes use of either CAP or BASE [36]. The popularity of agile development methods calls for techniques that offer higher scalability and performance to keep up with the ever-changing technical environment. In-memory database for high-update situations, like a website that displays everyone's "last active" time (for chat maybe). If users are performing some activity once every 40 seconds, then it will push RDBMS to limits with about 5000 simultaneous users for instance, when the numbers multiply by 10 [36].

7.2 Findings
The main problems found on the Flat file and RDBMS that were common to both database systems include security vulnerabilities, scalability limitations, availability of data regardless of network partition, timely propagation of changes to ensure consistency, performance bottlenecks, and the existence of a single point of failure. Owing to the rigid schema of the RDBMS, not all data structures can be represented and stored [36]. These challenges manifest as a result of the architectural constraints inherent in the databases. It was observed that these DBMS have some aspects that are still desirable for instance to achieve reliability and integrity. Completely doing away with the traditional databases in favor of the total adoption of the NoSQL also poses great challenges in our data management quest [34]. NoSQL has the challenges of not adequately catering to relational and transactional data. While giving cognizance to mission-critical data, transactional data, and a varied more cases where we seek to ensure reliability as a key aspect, NoSQL may not be ideal, calling for a revisit to the good old mature, tried and tested RDBMS. Owing to this scenario, both RDBMS and NoSQL are suited for different purposes and therefore cannot be absolute substitutes for each other [33].

7.3 Conclusion
The only feasible solution in the future for a single universal solution to cater to both relational and non-relational data would be an extension of the RDBMS to allow it to cater to non-relational data. Our future works will be an exploration of a solution that can assimilate the desirable features of the RDBMS and those of the NoSQL solutions in one database model.

James et al., (2017) compiled a review of the Hybrid Database System for Big Data Storage and Management

8. Description of the Research
Relational database systems have served as the actual storage systems for several years. However, within the last four years, there have been great resolutions in the computing world that have lessened the prevalence of relational databases which has led to an upthrust in the consideration of a new storage model called NoSQL [43]. This is because relational databases were never designed to store or manage such a volume of data, with high velocity and variety, unstructured data, or rapid growth. Enterprises are therefore turning to a new emerging storage approach called NoSQL for solutions to inherent challenges in big data. Most NoSLs scale horizontally with an increase in data volume and are also sufficiently flexible to hold partially structured and dispersed data collection.

In this work, we developed a hybrid database system using MongoDB and MySQL databases which are popular variants of NoSQL database and relational database systems respectively. Before data storage, data is categorized into structured data and unstructured data depending on the nature of the data. Unstructured data are stored and managed in the MongoDB database while storage and management of strictly structured data is carried out using MySQL database. Our hybrid system is such that, the databases making up the system can function separately for instance, our system could be used as a separate and complete MongoDB database. Instead of giving up the functions of relational database systems for the NoSQL database, we have developed an approach that offers the benefits of both systems in a single database system. NoSQL databases were developed to eliminate the limitations or drawbacks encountered in the use of relational databases especially in big data storage environments. As such, most of the drawbacks in the relational storage system form the strengths or advantages of NoSQL database systems.

8.1 Findings
We loaded big data in our hybrid database system in SQL mode, MongoDB mode, and also hybrid mode. The databases used for data storage and management vary depending on the mode in which the system runs. The output of the implemented hybrid database system for big data storage and management is discussed here; In SQL mode, data is stored and managed in MySQL database [37]. The system discards data in unstructured form since it cannot be stored in an SQL database. A screenshot of the load data operation performed by our hybrid system in SQL mode is given in Figure 7, a view of the database content shown in Figure 8 shows that unstructured data is discarded by our hybrid application in SQL mode. In MongoDB mode, storage and management of both structured and unstructured data is performed using the MongoDB database [37]. This stores both structured and unstructured data in MongoDB. Also in hybrid mode, data in structured form is stored and managed using SQL database while MongoDB is used to store and manage unstructured data.

8.2 Conclusion
We proposed a method that combines SQL database which belongs to the relational group of database systems and MongoDB is a NoSQL database to store and manage big data. With the result obtained it is understandable that our system can be used for storage and management of big eliminating the weaknesses in both databases.

Chapter Three

9. Research Methodology

9.1 Introduction
The rapid evolution of database management systems (DBMS) has been driven by the growing demand for handling large volumes of data, diverse data types, and real-time processing capabilities. Traditional relational database management systems (RDBMS) have long been the cornerstone of data management; however, the advent of big data has exposed their limitations in scalability and flexibility [38,39]. In response, new paradigms such as

NoSQL, NewSQL, and cloud-based databases have emerged, each promising to address these challenges more effectively.

This research aims to conduct a comprehensive comparative analysis of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases, focusing on their scalability, flexibility, and performance. Furthermore, it explores how these modern database technologies support real-time data processing and analytics, influence decision-making processes, and facilitate the development of new business models.

9.2 Research Design

Comparative Analysis of Traditional RDBMS, NoSQL, NewSQL, and Cloud-Based Databases in Terms of Scalability, Flexibility, Performance, and Impact on Decision-Making and Business Models.

This research adopts a comparative analysis method to comprehensively analyze the differences and impacts of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases.

9.3 Research Philosophy

The research philosophy underpinning this study on the comparative analysis of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases is rooted in pragmatism. Pragmatism is an appropriate philosophical approach for this research as it emphasizes the practical application of ideas by focusing on the problem at hand and using the most suitable methods available to investigate and understand complex phenomena.

- **Problem-Centric Focus:** Pragmatism is centered around solving practical problems. The research is driven by the need to address real-world challenges associated with the scalability, flexibility, and performance of various DBMS technologies. By focusing on these issues, the research aims to provide actionable insights and recommendations that can be directly applied in organizational contexts.
- **Flexibility and Adaptability:** Pragmatic research is inherently flexible and adaptable, allowing for the use of multiple methods and perspectives to explore the research questions [40-45]. This flexibility is crucial for investigating the complex and multifaceted nature of database management systems, which involve technical, operational, and strategic dimensions.
- **Emphasis on Practical Outcomes:** The ultimate goal of pragmatic research is to produce practical outcomes that can inform decision-making and lead to improvements in practice. This study aims to deliver valuable findings that organizations can use to enhance their data management strategies, optimize the performance of their database systems, and support the development of innovative business models.

9.4 Research Objectives

Compare the scalability, flexibility, and performance of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases. Evaluate the effectiveness of these database systems in supporting real-time data processing and analytics.

Analyze the influence of modern DBMS technologies on decision-making processes and the development of new business models. Identify challenges and trade-offs associated with transitioning from traditional RDBMS to modern database systems.

9.5 Data Collection Methods

An extensive review of academic journals, industry reports, and white papers was conducted to establish a theoretical framework and understand existing research on DBMS technologies.

Analysis was done by Examining the scalability, performance, flexibility, real-time processing capabilities, and impact on decision-making and business models within these organizations.

9.6 Data Analysis Methods

A comparative analysis was used to compare and contrast the findings from different DBMS types (traditional RDBMS, NoSQL, NewSQL, and cloud-based databases) to draw comprehensive conclusions on their relative advantages and disadvantages.

10. Chapter Four

- **Analysis**
- **Objective 1:** Comparison of Scalability, Flexibility, and Performance of Traditional RDBMS, NoSQL, NewSQL, and Cloud-Based Databases

Aspect	Traditional RDBMS	NoSQL	NewSQL	Cloud-Based DBMS
Scalability	Vertical scaling, limited	Horizontal scaling, excellent	Horizontal scaling with ACID	Elastic scaling, on-demand
Flexibility	Structured data, less flexible	Schema-less, highly flexible	Structured data, more flexible	Managed services, highly flexible
Performance	Optimized for transactions	High throughput, eventual consistency	High performance for transactions	Optimized performance, real-time

Each database management system type offers distinct advantages and trade-offs:

Traditional RDBMS are ideal for applications requiring complex transactions and strict ACID properties but face limitations in scalability and flexibility.

NoSQL databases provide excellent scalability and flexibility for large volumes of unstructured data, though they may compromise on consistency for performance.

NewSQL databases aim to offer the best of both RDBMS and NoSQL, delivering high performance and scalability with strong transactional integrity.

Cloud-based databases combine the benefits of managed services, elastic scalability, and optimized performance, making them suitable for a wide range of modern applications requiring real-time data processing and analytics.

• Objective 2

Evaluation of the Effectiveness of Database Systems in Supporting Real-Time Data Processing and Analytics

Criterion	**Traditional RDBMS**	**NoSQL**	**NewSQL**	**Cloud-Based DBMS**
Real-Time Data Processing	Moderate to Low	High	High	Very high
Latency	Higher latency due to ACID compliance	Low latency for high-throughput ops	Low latency With, ACID properties	Very low latency due to Cloud optimizations
Scalability for Real-Time	Limited scalability	Excellent scalability	Excellent scalability	Elastic scalability
Consistency	Strong Consistency (ACID)	Eventual consistency (varies by type)	Strong Consistency (ACID)	Configurable Consistency (depending on service)
Flexibility in Data Models	Limited to structured Data	High Flexibility (schema-less)	Moderate flexibility (structured data with more adaptability)	High flexibility (supports diverse data types)
Performance Under Load	Can degrade under high load	Maintains high performance under load	Maintains high performance under load	Optimized Performance under variable load
Analytics Integration	Moderate (Requires additional tools)	High (built-in analytics insome types)	High (optimized for mixed workloads)	Very high (Integrates with cloud analytics services)
Data Processing Frameworks	Compatible with traditional BI tools	Integrates with modern big data tools	Supports modern and traditional tools	Seamless integration with cloud-native tools
Cost Efficiency for Real-Time	High cost due to scaling limitations	Cost-effective for large datasets	Cost-effective with high performance	Cost-efficient due to pay-as-you-go model

In supporting real-time data processing and analytics:

Traditional RDBMS are less effective due to scalability and latency constraints.

NoSQL databases are highly effective, offering low latency and high throughput.

NewSQL databases provide a balanced approach with high performance and strong consistency.

Cloud-based DBMS offer the highest effectiveness due to their elastic scalability, low latency, and integration with cloud-native analytics tools.

Choosing the right database system depends on specific requirements, including the need for consistency, scalability, and the nature of the data and workloads.

• Objective 3

Influence of Modern DBMS Technologies on Decision-Making Processes and Business Models. Modern DBMS technologies significantly enhance decision-making processes and the development of new business models by offering:

NoSQL databases provide high flexibility, enabling quick adaptation to changing business environments and supporting real-time data processing [46,47].

NewSQL databases offer a balanced approach, combining the high performance and scalability of NoSQL with the transactional integrity of traditional RDBMS, making them suitable for both traditional and innovative business models.

Cloud-Based DBMS excel in scalability, flexibility, and integration with cloud services, supporting real-time analytics and diverse, innovative business models. Their cost-efficient and adaptable nature makes them ideal for rapidly evolving business landscapes.

Choosing the appropriate DBMS technology depends on the specific needs of the organization, including the volume and variety of data, the need for real-time processing, and the desired agility in decision-making and business model innovation.

• Objective 4

Challenges and Trade-Offs in Transitioning from Traditional RDBMS to Modern Database Systems

Challenges

- **Data Migration:** Transitioning large volumes of data can be complex, with risks of data loss and downtime.
- **Skill Requirements:** The need for new skills and training can be a significant hurdle.
- **System Integration:** Ensuring new systems work seamlessly with legacy applications can be problematic.
- **Cost:** High initial investments in new technologies and infrastructure are often required. Performance Optimization: Achieving optimal performance in new systems can involve significant tuning.
- **Security:** Ensuring data security and compliance during and after the transition poses challenges.

Transitioning from traditional RDBMS to modern database systems presents significant challenges and trade-offs. Organizations must carefully plan and execute this transition to balance immediate challenges with long-term benefits, such as improved scalability, flexibility, performance, and cost-efficiency [48-50]. Strategic planning, effective change management, and thorough training are essential to successfully navigating this complex landscape. Conclusion

10.1 Chapter Findings and Summary

The comparative analysis of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases reveals distinct characteristics and trade-offs in scalability, flexibility, performance, support for real-time data processing and analytics, influence on decision-making processes, and impact on business models.

Challenges include data migration complexity, skill requirements, system integration, cost, performance optimization, security, data consistency, vendor lock-in, maintenance and support, cultural resistance, backup and recovery, compliance and governance, and operational continuity.

Trade-offs involve data loss or corruption risks during migration, potential downtime, training costs and learning curve for new skills, compatibility issues, higher initial costs versus long-term savings, balancing performance with consistency and availability, dependency on specific vendors versus flexibility, initial setup complexity versus improved reliability, organizational inertia versus potential innovation.

In conclusion, the choice of database system depends on specific needs, including data structure, scalability requirements, performance considerations, and desired impact on decision-making and business models [51-53]. Each type of database system offers unique strengths and challenges, requiring careful consideration and strategic planning to maximize benefits and mitigate risks in transitioning and adoption.

11. Chapter Five

11.1 Conclusion

In conclusion, the comparative analysis of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases highlights the diverse landscape of database management systems, each offering unique strengths and challenges in scalability, flexibility, performance, and support for real-time data processing and analytics.

Traditional RDBMS, while well-suited for complex transactions, face limitations in scalability and flexibility (TechTarget, n.d.). NoSQL databases excel in scalability and flexibility, with a schema-less design ideal for handling large volumes of unstructured data (MongoDB, n.d.). NewSQL databases strike a balance between the scalability of NoSQL and the transactional integrity of traditional RDBMS (NuoDB, n.d.). Cloud-based databases offer elastic scalability, optimized performance, and integration with cloud-native services, making them highly effective for real-time data processing and analytics (Amazon Web Services, n.d.).

In terms of decision-making processes and business model innovation, NoSQL databases provide high flexibility, supporting new digital business models and rapid adaptation to market changes (DataStax, n.d.) [54-58]. NewSQL databases offer a balanced approach, supporting both traditional and innovative business models with high performance and strong consistency (Google Cloud, n.d.). Cloud-based databases enable diverse and scalable business models, offering high availability and integration with cloud-native services (Microsoft Azure, n.d.).

Transitioning from traditional RDBMS to modern database systems presents challenges such as data migration complexity, skill requirements, and system integration (Oracle, n.d.). However, the trade-offs include improved scalability, flexibility, and performance, as well as the ability to support real-time data processing and analytics (IBM, n.d.).

In conclusion, the choice of database system should align with specific organizational needs and goals. While traditional RDBMS may still be suitable for certain applications, NoSQL, NewSQL, and cloud-based databases offer compelling advantages for organizations seeking to enhance scalability, flexibility, and performance in their data management strategies. Successful adoption of modern DBMS technologies requires careful planning, strategic implementation, and ongoing evaluation to maximize benefits and minimize challenges in the evolving landscape of data management [59,60].

11.2 Implications

The comparative analysis of traditional RDBMS, NoSQL, NewSQL, and cloud-based databases has several implications for organizations:

• **Strategic Database Selection:** Organizations need to carefully evaluate their data management needs and select a database system that aligns with their scalability, flexibility, and performance requirements. This choice will have long-term implications for their ability to handle growing data volumes and support evolving business models.

• **Skill Development:** The adoption of modern database systems may require organizations to invest in training and skill development for their IT teams. This includes acquiring expertise in NoSQL, NewSQL, and cloud-based technologies to effectively manage and optimize these systems.

• **Data Integration and System Compatibility:** Organizations transitioning to modern database systems need to address integration challenges with legacy systems. This includes ensuring compatibility and seamless data exchange between different database systems and applications.

• **Cost Considerations:** While modern database systems offer benefits such as scalability and performance, organizations must carefully evaluate the costs associated with licensing, infrastructure, and ongoing maintenance. They should also consider the potential cost savings from improved efficiency and scalability.

• **Security and Compliance:** As organizations move towards modern database systems, they need to ensure that data security and compliance requirements are met. This includes implementing robust security measures and adhering to regulatory standards.

• **Business Model Innovation:** The adoption of modern database systems can enable organizations to innovate and develop new business models. These systems provide the flexibility and scalability needed to support innovative products and services.

In conclusion, the choice and adoption of a database system have wide-ranging implications for an organization's data management strategy, IT infrastructure, and business operations. By carefully evaluating their options and considering these implications, organizations can make informed decisions that align with their strategic objectives and future growth.

11.3 Recommendations

• **Evaluate Business Needs:** Before transitioning, conduct a thorough assessment of your organization's data management needs, including scalability requirements, data types, and performance expectations. This will help you choose the most suitable database system.

• **Develop a Transition Plan:** Create a detailed plan for transitioning to a modern database system, including data migration strategies, training for staff, and integration with existing systems. Consider using phased migration to minimize disruption.

• **Invest in Training:** Provide training for your IT team to ensure they have the necessary skills to manage and optimize the new database system. Consider partnering with external training providers or vendors for specialized training.

• **Address Integration Challenges:** Develop strategies to integrate the new database system with existing systems and applications. This may involve using middleware or APIs to facilitate data exchange [61,62].

• **Consider Cost Implications:** Evaluate the costs associated with licensing, infrastructure, and maintenance of the new database system. Compare these costs with potential savings and benefits to ensure a favorable cost-benefit ratio.

• **Prioritize Security and Compliance:** Implement robust security measures and ensure compliance with relevant regulations when transitioning to a new database system. Consider using encryption and access controls to protect sensitive data.

• **Explore Cloud-Based Solutions:** Consider leveraging cloud-based database services for scalability, flexibility, and cost efficiency. However, be mindful of potential vendor lock-in and ensure the chosen cloud provider meets your security and compliance requirements [63]. Following these recommendations, organizations can successfully transition to modern database systems and leverage their benefits to improve efficiency, scalability, and innovation in data management.

References

1. Bhattacharya, A., Lee, J., Agrawal, D., P. Krishna Reddy, Mukesh Mohania, Mondal, A., Goyal, V., & Rage Uday Kiran. (2022). *Database Systems for Advanced Applications. Springer Nature*.
2. Belcher, C. M. (2013). Fire Phenomena and the Earth System. *John Wiley & Sons*.
3. 2014 International Conference on Computer, Network. (2014). DEStech Publications, Inc.
4. Sitek, P., Pietranik, M., Krótkiewicz, M., & Srinilta, C. (Eds.). (2020). *Intelligent Information and Database Systems: 12th Asian Conference, ACIIDS 2020, Phuket, Thailand, March 23–26, 2020, Proceedings* (Vol. 1178). Springer Nature.
5. Baruah, A. (2002). Library database management. Gyan Publishing House.
6. Bajgoric, N. (2009). Always-On Enterprise Information Systems for Business Continuance: Technologies for Reliable and Scalable Operations. *IGI Global*.
7. Kannan, R. (Ed.). (2016). Managing and processing big data in cloud computing. *IGI Global*.
8. Paliszkiewicz, J. (2021). Management in the era of big data: issues and challenges. *CRC Press, Taylor & Francis Group*.
9. Lengnick-Hall, M., & Lengnick-Hall, C. (2003). Human resource management in the knowledge economy: New challenges, new roles, new capabilities. *Berrett-Koehler Publishers*.
10. Bonnefoy, P. Y., Chaize, E., Mansuy, R., Mehdi, T. A. Z. I., & Heckel, S. (2024). The Definitive Guide to Data Integration: Unlock the power of data integration to efficiently manage, transform, and analyze data. *Packt Publishing*.
11. Brahnam, S., & Jain, L. C. (2011). Virtual reality in psychotherapy, rehabilitation, and neurological assessment. In *Advanced Computational Intelligence Paradigms in Healthcare 6. Virtual Reality in Psychotherapy, Rehabilitation, and Assessment* (pp. 1-9). Berlin, Heidelberg: Springer Berlin Heidelberg.
12. Mahmood, Z. (2016). Data Science and Big Data Computing. *Data Science and Big Data Computing*.
13. Singh, N. P., & Gupta, C. S. (2014). Relational database management systems. *Abhishek Publications*.

14. Ravi, D. (2015). Recruitment and selection practices of IT companies in Andhra Pradesh–a study of select units. *Zenon Academic Publishing*.
15. Saueressig, G. (2002). An Analysis of the Current Development Stage of the eXtensible Markup Language (XML) and its Usage for Database Systems. diplom.
16. Baesens, B. (2014). Analytics in a big data world: The essential guide to data science and its applications. *John Wiley & Sons*.
17. Hurwitz, J., Kaufman, M., Bowles, A., Nugent, A., Kobielus, J. G., & Kowolenko, M. D. (2015). *Cognitive computing and big data analytics* (Vol. 288). Indianapolis: Wiley.
18. Hyman, J. A., Massaron, L., McFedries, P., Mueller, J. P., Reichental, J., Schmuller, J., & Taylor, A. G. (2024). Data Analytics & Visualization All-in-One For Dummies. *John Wiley & Sons*.
19. Popp, J., Tuchin, V. V., Chiou, A., & Heinemann, S. H. (Eds.). (2012). *Handbook of Biophotonics, Volume 3: Photonics in Pharmaceutics, Bioanalysis and Environmental Research* (Vol. 3). John Wiley & Sons.
20. Orăștean, R., Ogrean, C., Silvia Cristina Mărginean, & Springerlink (2021). Organizations and Performance in a Complex World: 26th *International Economic Conference of Sibiu (IECS)*. Springer International Publishing.
21. Bellini, A., & Bang, S. (2022, December). Barriers for data management as an enabler of circular economy: an exploratory study of the Norwegian AEC-industry. In *IOP Conference Series: Earth and Environmental Science* (Vol. 1122, No. 1, p. 012047). IOP Publishing.
22. Boicea, A., Radulescu, F., & Agapin, L. I. (2012). MongoDB vs Oracle--database comparison. In *2012 third international conference on emerging intelligent data and web technologies* (pp. 330-335). IEEE.
23. McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D. J., & Barton, D. (2012). Big data: the management revolution. *Harvard business review, 90*(10), 60-68.
24. Nayak, A., Poriya, A., & Poojary, D. (2013). Type of NOSQL databases and its comparison with relational databases. *International Journal of Applied Information Systems, 5*(4), 16-19.
25. Big Data definition in the Gartner IT Glossary.
26. Anderson, C., Dencik, J., Marshall, A., & Teer, R. (2022). The co-evolution of data and the modern enterprise. *Strategy & Leadership, 50*(6), 33-40.
27. Tauro, C. J., Patil, B. R., & Prashanth, K. R. (2013). A comparative analysis of different nosql databases on data model, query model and replication model. In *Proceedings of the International Conference on ERCICA*.
28. Boyd, D., & Crawford, K. (2012). Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, communication & society, 15*(5), 662-679.
29. Draft NIST Big Data Interoperability Framework: Volume 1, Definitions‖, NIST Special Publication 1500-1, April 6, 2015
30. Geoinformatics, Department of Civil Engineering, IIT Kanpur, Retrieved from http://gi.iitk.ac.in/gi/geoinformatics [28] A. Banafa, ―The Future of Big Data and Analytics‖, School of Business and Information Technology, March 2014, Retrieved from http://www.kaplanuniversity.edu/information-technology/articles/futureof-big-data-analytics.aspx
31. Buhl, H. U., Röglinger, M., Moser, F., & Heidemann, J. (2013). Big data: a fashionable topic with (out) sustainable relevance for research and practice?. *Business & Information Systems Engineering, 5,* 65-69.
32. Center, I. I. (2012). Planning guide: Getting started with hadoop. Steps IT Managers can take to move forward with big data analytics.
33. Gantz, J., Reinsel, D. (2011). Extracting value from chaos‖. IDC view 1142.
34. Gray, J. (1981). The transaction concept: Virtues and limitations. In *VLDB* (Vol. 81, pp. 144-154).
35. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., and A.H. Byers. (2011). "Big data: The next frontier for innovation, competition, and productivity". McKinsey Global Institute.
36. Pokorny, J. (2011). NoSQL databases: a step to database scalability in web environment. In *Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services* (pp. 278-283).
37. Bakshi, K. (2012). Considerations for big data: Architecture and approach. In *2012 IEEE aerospace conference* (pp. 1-7). IEEE.
38. Fanning, K., & Drogt, E. (2014). Big data: new opportunities for M&A. *Journal of Corporate Accounting & Finance, 25*(2), 27-34.
39. Grolinger, K., Higashino, W. A., Tiwari, A., & Capretz, M. A. (2013). Data management in cloud environments: NoSQL and NewSQL data stores. *Journal of Cloud Computing: advances, systems and applications, 2*, 1-24.
40. Kurniawan, Maulana, A., & Iskandar, Y. (2023). The effect of technology adaptation and government financial support on sustainable performance of MSMEs during the COVID-19 pandemic. *Cogent Business & Management, 10*(1), 2177400.
41. Zyp, K. (2010). NoSQL Architecture. SitePen Blog. May, 11, 11.
42. Ndlela, L. T., & Du Toit, A. S. (2001). Establishing a knowledge management programme for competitive advantage in an enterprise. *International journal of information management, 21*(2), 151-165.
43. Wu, L., Yuan, L., & You, J. (2015). Survey of large-scale data management systems for big data applications. *Journal of computer science and technology, 30*(1), 163-183.
44. Han, M., & Jeon, J. (2023). Roadmap incorporating data management perspective for platform business model innovation. *Sustainability, 15*(4), 3151.
45. Indrawan-Santiago, M. (2012). Database research: Are we at a crossroad? Reflection on NoSQL. In *2012 15th International Conference on Network-Based Information Systems* (pp. 45-51). IEEE.
46. Smith, M., Szongott, C., Henne, B., & Von Voigt, G. (2012). Big data privacy issues in public social media. In *2012 6th IEEE international conference on digital ecosystems and technologies (DEST)* (pp. 1-6). IEEE.

47. Walker, M. (2012). Data Veracity‖. *Data Science Central, November 28*.
48. Freitag, P. M., & Brettel, M. (2017). Building up dynamic capabilities for the digital age. In *Academy of Management Proceedings* (Vol. 2017, No. 1, p. 12890). Briarcliff Manor, NY 10510: Academy of Management.
49. Raj, P., & Vijayakumar, P. (2022). Describing the IoT data analytics methods and platforms. Streaming Anal. *Concepts, Archit. Platforms, Use Cases Appl, 44*, 191.
50. Russom, P. (2011). Big data analytics. *TDWI best practices report, fourth quarter, 19*(4), 1-34.
51. Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., & Kostrzewa, D. (Eds.). (2016). Beyond Databases, Architectures and Structures. Advanced Technologies for Data Mining and Knowledge Discovery: *12th International Conference, BDAS 2016, Ustroń, Poland, May 31-June 3, 2016, Proceedings* (Vol. 613). Springer.
52. Burtica, R., Mocanu, E. M., Andreica, M. I., & Ţăpuş, N. (2012). Practical application and evaluation of no-SQL databases in Cloud Computing. In *2012 IEEE International Systems Conference SysCon 2012* (pp. 1-6). IEEE.
53. Weiss, R., & Zgorski, L. J. (2012). Obama administration unveils 'big data'initiative: announces $200 million in new R&D investments. Office of Science and Technology Policy Executive Office of the President.
54. Schneider, R. D. (2012). Hadoop for dummies. Special Edition). Mississauga, Canada: *John Wiley & Sons*.
55. Rackspace, "NoSQL Ecosystem," Rackspace Blog.
56. Abiteboul, S. (1997). Querying semi-structured data. In Database Theory—ICDT'97: 6th International Conference Delphi, Greece, January 8–10, 1997 Proceedings 6 (pp. 1-18). Springer Berlin Heidelberg.
57. Siewert, S. B. (2013). Big data in the cloud. Data velocity, volume, variety, veracity, 4-21.
58. Connolly, S. (2012). 7 Key Drivers for the Big Data Market‖, May 14.
59. Boguda, S. K. (2020). The Revolutionary Paradigm of Enterprise Applications through the Lens of Data and Technology.
60. Madden, S. (2012). From databases to big data. *IEEE Internet Computing*. Volume 16, Issue 3.
61. Iskandar, Y. (2022). The role of digital innovation as A MSME business resilience strategy during the covid pandemic. Budapest International Research and Critics Institute-Journal (*BIRCI-Journal), 5*(2).
62. Iskandar, Y., Heliani, H., Jaman, U. B., & Ardhiyansyah, A. (2023). Analyzing the Relationship Between Technology Adoption and Business Performance in the Digital Age in SMEs in Indonesia. *Eastasouth Proceeding of Nature, Science, and Technology, 1*(01), 43-53.
63. Ma, Z., & Yan, L. (Eds.). (2019). Emerging technologies and applications in data processing and management