A logo with blue and orange letters

AI-generated content may be incorrect.

**Database Search and Reporting**

**Done by: Ikhlas Said Al Khusaibi.**

**Date: 16/6/2025.**

1. **Introduction:**

**🔍Objective overview:**

This report covers the important roles in database projects, basic database principles, a structured comparison of various systems, the benefits of Database Management Systems (DBMS), and different database types including their relationship to cloud storage.

1. **Comparison Assignment: Flat File Systems vs. Relational Databases**

|  |  |  |
| --- | --- | --- |
| Feature | Flat File Systems | Relational Databases (RDBMS) |
| Structure | Typically, simple text formats like CSV or TXT. Data is stored in basic tables or standalone files without clear links. | Highly structured format with data organized into rows and columns across related tables using defined keys. |
| Data Redundancy | Data is often repeated across files, leading to inconsistencies and inefficient storage. | Redundancy is minimized through normalization and relational design, ensuring consistency and efficient storage. |
| Relationships | Relationships aren’t clearly defined and must be handled within the application logic. | Relationships are clearly defined using primary and foreign keys, supporting complex data structures and integrity. |
| Example Usage | |  | | --- | |  |  |  | | --- | | Best for small-scale applications like configuration files, log storage, or simple datasets. | | Suited for enterprise systems, e-commerce sites, banking apps, and other use cases requiring structured, relational data. |
| Drawbacks | Poor data integrity and consistency | - Setting up and maintaining the system can be complicated- Large-scale implementations often need a dedicated database expert - Scaling to handle massive or unstructured data can be difficult. |

1. **DBMS Advantages – Mind Map (Textual Representation)**

A Database Management System (DBMS) provides several key benefits compared to conventional file systems for data management.

* **Integrity:** Data validation and consistency rules.
* **Security:** Access control and encryption.
* **Backup:** Disaster recovery.
* **Data sharing:** Centralized repository and controlled access.
* **Reduce redundancy:** Single source and consistency.
* **Concurrency:** Multi-user access.

1. **Roles in a Database System** **👥**

* **System Analyst**: Acts as a contact between business users and technical teams. They gather user requirements and analyze current systems to identify improvements, converting business needs into technical specifications.
* **Database Designer**: Converts requirements into database structures. They create ER diagrams, define schemas (conceptual, logical, and physical), and ensure the design supports performance and data integrity.
* **Database Developer**: Builds the database based on the design. They write scripts to create and manage database objects (tables, views, triggers) and implement logic through procedures and functions.
* **Database Administrator (DBA)**: Manages the database environment. Their responsibilities include setup, security, performance change, backups, and ensuring continuous availability.
* **Application Developer**: Develops applications that interact with the database. They write application code to manipulate data, relying on the database structure and logic provided by designers and DBAs.
* **BI Developer**: Specializes in analytics and reporting. They perform ETL processes, design dashboards and reports, and help organizations make informed decisions using visualized data.

1. **Types of Databases**

**🔁Relational vs. Non-Relational Databases:**

|  |  |  |
| --- | --- | --- |
| Attribute | Relational Databases (SQL) | Non-Relational Databases (NoSQL) |
| Description | Data is organized in tables with rows and columns; relationships are defined using primary and foreign keys; uses SQL. | Data is stored in flexible formats like documents, key-value pairs, graphs, or wide-columns; schema-less or flexible schema. |
| Characteristic | ACID compliant, strong consistency, rigid schema, ideal for structured data. | Flexible schema, horizontal scalability, eventual consistency (in many cases), support unstructured/semi-structured data. |
| Examples | MySQL, PostgreSQL, Oracle Database, Microsoft SQL Server | MongoDB (Document), Cassandra (Column-Family), Redis (Key-Value), Neo4j (Graph) |
| Use Case | - **E-commerce:** orders, inventory, transactions - **Banking:** account and transaction management - **ERP:** finance, HR, supply chain | - **MongoDB**: CMS, mobile apps, real-time analytics - **Cassandra**: messaging, IoT, analytics - **Redis**: caching, sessions - **Neo4j**: social networks, fraud detection |

**🔁Centralized vs. Distributed vs. Cloud Databases**

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Centralized Database | Distributed Database | Cloud Database |
| Description | All data is stored and accessed from a single server or location. | Data is spread across multiple connected computers or nodes, often in different locations. | Hosted on cloud platforms and managed by providers like AWS, Azure, or Google Cloud. |
| Characteristic | Easy to manage, cost-effective to set up, but has a single point of failure. | Offers high availability, fault tolerance, better performance, and scalability. | Scalable on demand, pay-per-use pricing, managed services, resilient with disaster recovery. |
| Use Case | Small to mid-sized businesses, department-level systems, single-server web apps. | Global organizations, large web platforms, real-time systems, content delivery networks. | Startups, variable workloads, globally accessed apps, backup/recovery-focused deployments. |

1. **Cloud Storage and Databases**

**☁️Cloud Storage and Its Role in Database Functionality**

**Cloud storage** is a data storage model where information is kept in virtualized pools hosted by third-party providers across multiple servers. Users access this storage via the internet without managing physical infrastructure.

**How it supports databases:**

* **Scalability**: Easily adjust storage size based on demand without hardware changes.
* **Reliability**: Data is redundantly stored across various locations, reducing the risk of loss.
* **Accessibility**: Enables global data access and remote database connectivity.
* **Cost Efficiency**: Pay-as-you-go model eliminates hardware costs and lowers maintenance expenses.
* **Backup & Recovery**: Supports automated backups and rapid data restoration in case of failures.

**🧾Advantages and Disadvantages of using cloud-based database**

**Advantages and Disadvantages of using cloud-based database details:**

**✅ Advantages**

* **Scalable & Elastic**: Resources can be adjusted automatically based on demand.
* **Highly Available & Durable**: Redundancy and failover ensure continuous service and data protection.
* **Managed Infrastructure**: Cloud providers handle updates, backups, and maintenance.
* **Cost-Efficient**: Pay-as-you-go model reduces upfront costs and allows usage-based billing.
* **Global Accessibility**: Databases can be deployed close to users, lowering latency.
* **Strong Security**: Providers offer encryption, compliance support, and advanced protection.

**⚠️ Disadvantages**

* **Vendor Lock-In**: Switching providers can be difficult and expensive.
* **Security Responsibility**: Users must still manage data protection and compliance.
* **Performance Fluctuations**: Shared resources can lead to inconsistent speed.
* **Cost Complexity**: Billing can be unpredictable without careful monitoring.
* **Internet Dependence**: Access requires a stable connection.
* **Egress Costs**: Transferring data out of the cloud can be expensive.

1. **Conclusion**

This report summarizes key concepts in database systems, comparing flat file systems with relational databases and exploring different types like relational, non-relational, centralized, distributed, and cloud-based databases. It outlines the main roles in database projects and the benefits of using DBMS, such as improved data integrity, security, and efficiency.

Cloud-based databases are highlighted for their scalability, availability, and cost-effectiveness, though challenges like vendor lock-in and internet dependence are noted. Overall, the report helps guide the selection of appropriate database solutions based on specific business and technical needs.