Contents

[Table of Figures 1](#_Toc141998225)

[Introduction 2](#_Toc141998226)

[Database Technologies 2](#_Toc141998227)

[Relational Database Management Systems (RDBMS) 2](#_Toc141998228)

[NoSQL – Not Only SQL 4](#_Toc141998229)

[Hierarchical Databases: 4](#_Toc141998230)

[Document Database 4](#_Toc141998231)

[Key-Value Database 5](#_Toc141998232)

[Graph Database 6](#_Toc141998233)

[Wide-Column Database 7](#_Toc141998234)

[Data Lakes 7](#_Toc141998235)

[Appendices 7](#_Toc141998236)

[Appendix A: 7](#_Toc141998237)

[Figure 1 – RDBMS Table 7](#_Toc141998238)

[Figure 2 – Stack Overflow Database Environments Survey: 7](#_Toc141998239)

[Figure 3 – ACID Principles: 8](#_Toc141998240)

[Figure 4 – Hierarchical Data Tree 8](#_Toc141998241)

[Figure 5 – Document Model Example 8](#_Toc141998242)

[Figure 6 – Key-Value Database Example 8](#_Toc141998243)

[Figure 7 – Shortest Path Key-Value Data Access 8](#_Toc141998244)

[Figure 8 – Graph Database example Schema 9](#_Toc141998245)

[Figure 9 – Wide Column Database Example 9](#_Toc141998246)

[Appendix B: 9](#_Toc141998247)

[Figure 1 – RDBMS Table: 9](#_Toc141998248)

[Figure 2 – Hierarchical Data Example: 9](#_Toc141998249)

[References 10](#_Toc141998250)

[Glossary 11](#_Toc141998251)

# Table of Figures

[Figure 1 Example RDBMS SQL Table Relations (W3Resource 2022) 2](#_Toc141821737)

[Figure 2 - Database Environments used by Professional Developers - Stack Overflow Survey 2023 3](#_Toc141821738)

[Figure 3 - ACID Principles 3](#_Toc141821739)

[Figure 4 - Hierarchical Data Structure - MariaDB 2015 4](#_Toc141821740)

# Introduction

This report uses a variety of specialized terminology and abbreviations. Explanations are defined in the Glossary.

# Database Technologies

## Relational Database Management Systems (RDBMS)

Inferred from the name, the RDBMS uses a structure based on the relational model proposed by E.F. Codd in 1970 (Connolly 2015 Part 2 – Chapter 4), to allow us to identify and access data attributes via table-based (Appendix B - Figure 1) relations using primary and foreign keys.

A screenshot of a computer

Description automatically generated

Figure 1 Example RDBMS SQL Table Relations (W3Resource 2022)

They are designed to represent complex data schemas and minimize data redundancy through normalization whilst offering performant transactions using SQL[[1]](#endnote-1).

RDBMS vendors, such as *PostgreSQL, MySQL & SQLite* are amongst the most common databases used by professional developers, evidenced below.

A screenshot of a graph

Description automatically generated

Figure 2 - Database Environments used by Professional Developers - Stack Overflow Survey 2023

An excellent reason to use an RDBMS’ is when your transactions must adhere to the ACID principles[[2]](#endnote-2) which most vendors support implicitly, and if your organisation has complex querying requirements.

A screenshot of a computer

Description automatically generated

Figure 3 - ACID Principles

Any system that prefers reliable and consistent data storage with complex structure over performance will favour an RDBMS, such as:

* Banking systems
* Education sector systems
* Complex online stores

While RDMS’ advantages give them the number 1 spot in industry, their pros can be their greatest limitation for certain requirements:

* ACID inherently reduces performance, limiting their adoption by platforms like YouTube (Shivang, 2019).
* Unsuited for unstructured/semi-unstructured data such as JSON [[3]](#endnote-3)documents.
* Extreme complexity can lead to a poor representation of “real-world” entities & challenging schema updates (Connolly 2015, Part 2 – Chapter 9).

## NoSQL – Not Only SQL

### Hierarchical Databases:

Hierarchical Databases, like JSON, display data in a tree-form[[4]](#endnote-4) with unidirectional parent-child links (Appendix B - Figure 2).

A diagram of a network

Description automatically generated

Figure 4 - Hierarchical Data Structure - MariaDB 2015

Highly efficient for read/write operations due to their structure, they excel in explicit hierarchical relationships like File Systems (MariaDB, 2015). Their limitation lies in an inability to support multi-dimensional or many-to-many relationships.

### Document Database

Despite the nomenclature, they don’t store entities like Spreadsheets or Word documents. Instead, it is structured formats like JSON, BSON[[5]](#endnote-5) or XML[[6]](#endnote-6) (Hoffer, 2019).

Each named, key-addressed record stores its data as a “value”. Unlike hierarchical databases, they facilitate many-to-many relationships by referencing another document’s key.

A close-up of a computer code

Description automatically generated

Figure 5 Document Data Model Representation (Pore, 2018)

They are scalable, can handle complex data and have flexible schemas, making them great for CMS’[[7]](#endnote-7) using *MongoDB* or *Amazon DynamoDB* however they lack proper structure and complex queries can be slow.

### Key-Value Database

KVP [[8]](#endnote-8) databases, fully non-relational, are favored by Twitter (Pandori, 2022) & LinkedIn for rapid lookup in distributed cloud environments.

A screenshot of a computer

Description automatically generated

Figure 6 Example of a Key Value database. (Redis 2023)

Utilizing a hash-table[[9]](#endnote-9) to store data, they offer near instantaneous access but don’t support querying the unstructured values (Hoffer, 2019).

A graph with different colored bars

Description automatically generated

Figure 7 DB Providers & DB Types Read Time for Related Nodes (ArangoDB 2018)

Different from other databases, KVDs excel in speed and scalability, yet lack query versatility.

### Graph Database

Employing graph theory[[10]](#endnote-10) to store, map, and query relationships, they’re advantageous for interconnected relationships, making them ideal for recommendation engines[[11]](#endnote-11), advanced analytics (Neo4J, n.d.) and fraud detection.

A diagram of a person

Description automatically generated

Figure 8 A simple Graph schema example (Morgante 2020)

They’re unique in NoSQL realm for their relational nature, but struggle with scalability and require complex data modelling.

### Wide-Column Database

Excellent for large datasets with simple querying requirements, favoring tables with numerous, varying column structures (Hoffer, 2019) over table-joins. However, they have a steep learning curve and may struggle with complex querying.

A diagram of a personal information

Description automatically generated

Figure 9 A simple example of a Wide-Column Database

They’re often found in data-warehousing for analytics, aggregation [[12]](#endnote-12)and data mining[[13]](#endnote-13).

### Data Lakes

A data lake is a centralized repository that allows storage of structured and unstructured data at any scale. It enables diverse analytic methods like machine learning. Unlike traditional systems, data lakes store raw data, offering flexibility but also demanding robust governance for efficient usage.

# Appendices

## Appendix A:

### Figure 1 – RDBMS Table

A simple representation of a table in a Relational Database Management System.

Adapted from “RDBMS Table Terminologies”, a blogpost by Wentz Wu. 8/07/2019.

### Figure 2 – Stack Overflow Database Environments Survey:

A chart displaying database environments used by professional developers (60,369 respondents). Answers are multiple choice to encapsulate total usage rather than the greatest usage of a specific vender.

Adapted from the 2023 Developer Survey by Stack Overflow.

### Figure 3 – ACID Principles:

A graphic showing an overview of the ACID Principles and their definitions.

Adapted from an independent blogpost by Dave Pinal. 9/12/2007.

### Figure 4 – Hierarchical Data Tree

A simple graphic showing the tree-like structure of hierarchical databases.

Adapted from an article from MariaDB. 06/06/2015.

### Figure 5 – Document Model Example

An image that represents a collection of documents within a document database. Records can contain different data if the overall structure is adhered to.

Adapted from an article from Akshay Pore, 16/2/2018.

### Figure 6 – Key-Value Database Example

An image that shows a human readable example of a Key-Value data store. Using a Name as a Key & a phone number as a Value

Adapted from an article by Redis DB, 06/07/2023.

### Figure 7 – Shortest Path Key-Value Data Access

A chart that displays the read time of related nodes in an inter-connected Social Media database context. ArangoDB(Rocks) is a Key-Value database.

Adapted from an article of benchmarks by ArangoDB, 14/02/2018.

### Figure 8 – Graph Database example Schema

This chart shows a simple graph database, each line represents a relationship between nodes.

Adapted from an article by Victor Morgante on Towards Data Science, 16/09/2020.

### Figure 9 – Wide Column Database Example

An example of a Wide Column store with an example of a User and the data for said user’s information across “column families”.

Adapted from an article from Database Town 29/1/2023.

## Appendix B:

### Figure 1 – RDBMS Table:

A diagram of a table

Description automatically generated

A simple example of an RDBMS table and a visual representation of definitions associated with them.

Adapted from an example from W3Resource’s website. 19/08/2022.

### Figure 2 – Hierarchical Data Example:

A screenshot of a computer screen

Description automatically generated

A visual representation of a Hierarchical data structure.

Organization is the Root of the hierarchy, with Departments & Employees as nested child entities. This demonstrates the limitation of the hierarchical data structure, as children can only have a single parent node.

# References

Akshay Pore, 16/02/2018. NoSQL Data Architecture & Data Governance: Everything You Need To Know [article] Available at: <https://www.dataversity.net/nosql-data-architecture-data-governance-everything-need-know/> [Accessed 01/08/2023]

ArangoDB, 14/02/2018. NoSQL Performance Benchmark 2018 – MongoDB, PostgreSQL, OrientDB, Neo4j and ArangoDB. [article] Available at: <https://www.arangodb.com/2018/02/nosql-performance-benchmark-2018-mongodb-postgresql-orientdb-neo4j-arangodb/> [Accessed 03/08/2023]

Connolly, Thomas, 17/4/2015. Database systems: a practical approach to design. Part 2 Chapter 4 – The Relational Model, 6th Edition. Pearson Education. Rotherham, England. [Accessed 31/07/2023]

Connolly, Thomas, 17/4/2015. Database systems: a practical approach to design. Part 2 Chapter 9 – Object-Relational DBMS, 6th Edition. Pearson Education. Rotherham, England. [Accessed 01/08/2023]

Database Town, 29/1/2023. Wide Column Database (Use Cases, Examples, Advantages & Disadvantages) [article] Available at: <https://databasetown.com/wide-column-database-use-cases/#wide-column-database-examples> [Accessed 03/08/2023]

Hoffer, Jeff. Modern database management. Part IV Chapter 10 – Big Data Technologies, page 485. Global Edition. Pearson Education. Rotherham, England. [Accessed 01/08/2023]

MariaDB, 06/06/2015. Understanding the Hierarchical Database Model. [article] Available at: <https://mariadb.com/kb/en/understanding-the-hierarchical-database-model/> [Accessed 01/08/2023]

Neo4J, 04/02/2015. Why Graph Database? [online] Available at: <https://neo4j.com/why-graph-databases/> [Accessed 03/08/2023]

Redis, 6/07/2023. What is a Key-Value Database. [article] Available at: <https://redis.com/nosql/key-value-databases/> [Accessed 02/08/2023]

Stack Overflow, June 2023. 2023 Developer Survey. [survey] Available at: <https://survey.stackoverflow.co/2023/#section-most-popular-technologies-databases> [Accessed 01/08/2023]

Sahib Pandori, 28/02/2022. Data transfer in Manhattan using RocksDB – Twitter Blogs. [blog] Available at: <https://blog.twitter.com/engineering/en_us/topics/infrastructure/2022/data-transfer-in-manhattan-using-rocksdb> [Accessed 02/08/2023]

Shivang, 15/12/2019. YouTube Database and how does it store so many videos without running out of storage space. [article] Available at: <https://scaleyourapp.com/youtube-database-how-does-it-store-so-many-videos-without-running-out-of-storage-space/> [Accessed 01/08/2023]

Pinal, Dave, 9/12/2007. SQL Server – ACID (Atomicity, Consistency, Isolation, Durability) [blog] Available at: <https://blog.sqlauthority.com/2007/12/09/sql-server-acid-atomicity-consistency-isolation-durability/> [Accessed 01/08/2023]

Victor Morgante, 16/09/2020. What is a graph database? Towards Data Science. [article] Available at: <https://towardsdatascience.com/what-is-a-graph-database-249cd7fdf24d> [Accessed 03/08/2023]

Wentz Wu, 8/07/2019, RDBMS Table Terminologies, RDBMS Table. [online] Available at: <https://wentzwu.com/2019/07/08/rdbms-table-terminologies/> [Accessed 31/07/2023]

W3Resources, 19/08/2022, SQL Table. [online] Available at: <https://www.w3resource.com/sql/sql-table.php> [Accessed 01/08/2023]

W3Schools, n.d. Introduction to XML. [online] Available at: <https://www.w3schools.com/xml/xml_whatis.asp> [Accessed 01/08/2023]

# Glossary

1. SQL – Structured Query Language

   The standard language of accessing and manipulating Relational Database Management Systems. [↑](#endnote-ref-1)
2. ACID Principles:

   Atomicity, Consistency, Isolation & Durability. A set of principles applied to database transactions to enhance the reliability of data, avoid stateful concurrency issues in transactions and eliminate data loss through critical outages. [↑](#endnote-ref-2)
3. JSON – JavaScript Object Notation

   A lightweight format of data often sent across HTTP requests back and forth between a browser and server. Heavily adopted due to the format matching that of Object-Oriented Programming Languages, such as C# and JavaScript, making serialization of data easy for communication. [↑](#endnote-ref-3)
4. Tree Structure

   A data structure that contains a single root node that can have a recursive number of child elements with only one parent. Called a “Tree” due to its triangular visual representation and because each child node can be referred to as a branch, where each branch has its own branches. [↑](#endnote-ref-4)
5. BSON – Binary JSON

   A binary representation of JSON, often storing metadata about the type & length of data. [↑](#endnote-ref-5)
6. XML – Extensible Markup Language

   A file format for storing, transmitting, and reconstructing arbitrary data. It defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. (W3Schools, n.d.) [↑](#endnote-ref-6)
7. CMS – Content Management System

   A software system that creates, manages, and modifies digital content on a website or application.] [↑](#endnote-ref-7)
8. KVP – Key-Value Pair

   A name & value pairing. Sometimes called an attribute-value pair. These are fundamental for data representation in computing systems and applications. [↑](#endnote-ref-8)
9. Hash Table

   In computing, a hash map is an abstract data type that is used to store an unordered collection of key-value pairs. Each key is unique which offers extremely fast data retrieval if you have the address of the key ahead of time. [↑](#endnote-ref-9)
10. Graph Theory

    Graph theory is a mathematical field studying graph structures, modeling pairwise relations between objects, characterized by vertices (nodes) and edges (connections) [↑](#endnote-ref-10)
11. Recommendation Engine

    This is a system that suggest products, services or information to users based on analysis of their behaviour. [↑](#endnote-ref-11)
12. Aggregation

    In data warehousing, aggregation refers to the process of combining or grouping data together to provide summarized, more digestible information. [↑](#endnote-ref-12)
13. Data Mining

    Applying a statistical method to create measurable, analysed data, revealing patterns or trends, contributing to model development and predictive analytics. [↑](#endnote-ref-13)