Contents

[Table of Figures 1](#_Toc141812015)

[Introduction 1](#_Toc141812016)

[Database Technologies 1](#_Toc141812017)

[Appendices 3](#_Toc141812018)

[References 4](#_Toc141812019)

[Glossary 4](#_Toc141812020)

# Table of Figures

[Figure 1 RDBMS Table Terminologies (Wentz Wu 2019) 1](#_Toc141729506)

# 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), to allow us to identify and access data attributes via table-based relations using primary and foreign keys.

A diagram of a table

Description automatically generated

Figure 1 RDBMS Table Terminologies (Wentz Wu 2019)

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, shown in Figure 2.

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. Example systems are complex online stores, banking systems and any system that prefers reliable, structured, and consistent data storage over performance.

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.

Complex because of normalization

Challenging to change

Poor support for complex data types like documents.

**NoSQL**

Hierarchical Databases

Network Database

Document

Graph

Key-Value

Wide-Column

Object Oriented Databases

Time Series Databases

# Appendices

Figure 1:

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:

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.

# References

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

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

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]

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]

# 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)