

Cuadro Comparativo de Bases de Datos Relacionales, No Relacionales y Multimodelo

En el ámbito de la gestión y almacenamiento de información, las bases de datos cumplen un papel fundamental para el funcionamiento de sistemas informáticos y aplicaciones modernas. Existen diversos tipos de bases de datos, siendo las más representativas las relacionales, no relacionales y multimodelo. Cada una de ellas posee características, ventajas y limitaciones específicas que las hacen más adecuadas para determinados escenarios. El siguiente cuadro comparativo presenta una síntesis de sus principales diferencias en cuanto a estructura, lenguaje, rendimiento, escalabilidad y uso típico.

Características	Base de Datos Relacional (SQL)	Base de Datos No Relacional (NoSQL)	Base de Datos Multimodelo
Modelo de datos	Basado en tablas (filas y columnas).	Basado en documentos, clave-valor, grafos o columnas.	Combina varios modelos (tablas, documentos, grafos, etc.).
Estructura	Estructura fija y bien definida mediante esquemas.	Estructura flexible, sin esquema fijo (schema-less).	Puede usar esquemas fijos o flexibles según el modelo.
Lenguaje de consulta	SQL (Structured Query Language).	Depende del tipo (ej. MongoDB usa consultas JSON, Neo4j usa Cypher).	Soporta múltiples lenguajes según el modelo usado.
Escalabilidad	Escalabilidad vertical (aumentar potencia de un solo servidor).	Escalabilidad horizontal (añadir más servidores).	Puede escalar vertical y horizontalmente.
Integridad de datos	Muy alta (cumple reglas ACID).	Variable (según el tipo, suele priorizar velocidad y disponibilidad).	Puede ofrecer ACID y también flexibilidad tipo NoSQL.
Rendimiento	Excelente para transacciones complejas y datos estructurados.	Excelente para grandes volúmenes de datos no estructurados o distribuidos.	Buen rendimiento en entornos mixtos y de gran escala.

Ejemplos	MySQL, PostgreSQL, Oracle, SQL Server.	MongoDB, Cassandra, Redis, Neo4j, CouchDB.	ArangoDB, Azure Cosmos DB, OrientDB.
Ventajas	Consistencia, integridad y facilidad de análisis con SQL.	Flexibilidad, rapidez y escalabilidad en grandes volúmenes.	Versatilidad, soporte para múltiples modelos de datos en una sola plataforma.
Desventajas	Poca flexibilidad para cambios de estructura y menor rendimiento en datos no estructurados.	Menor consistencia y estandarización; cada sistema tiene su propia forma de consulta.	Mayor complejidad de gestión y configuración.
Uso típico	Sistemas financieros, ERP, CRM, aplicaciones empresariales.	Big Data, redes sociales, IoT, contenido web.	Aplicaciones híbridas, análisis complejos, sistemas con datos heterogéneos.