

<codoa
codo/>

Temas a tratar Que es

mongo DB

Limitaciones Base de datos relacionalesBase de

datos NoSQL

SQL vs NoSQL

Agencia de
Aprendizaje
a lo largo
de la vida

¿Qué es Mongo DB?



mongoDB®

MongoDB es un sistema de base de datos NoSQL, orientado a documentos y de código abierto.

En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos JSON con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value
← field: value
← field: value
← field: value

¿Qué es un JSON?

Sus siglas son por Javascript Object Notation, se trata de un formato para guardar e intercambiar información que cualquiera pueda leer. Los archivos del tipo json contienen solo texto y usan la extensión .json, JSON es un formato que almacena información estructurada y se utiliza principalmente para transferir datos entre un servidor y un cliente.

Hay dos elementos centrales en un objeto del tipo JSON: claves (key) valores (values).

Las key deben ser cadenas de caracteres (string). Como su nombre en español lo indica, estas contienen una secuencia de caracteres rodeados de comillas.

Los valores son de un tipo de datos JSON válido. Puede tener la forma de un arreglo(array), objeto, cadena, booleano, número o nulo.

Un objeto del tipo JSON comienza y termina con llaves {}. Puede tener dos o más pares de claves/valor dentro, con una coma para separarlos, así como también cada key es seguida por dos puntos para distinguir el valor.

El modelo de documentos de MongoDB resulta muy fácil de aprender y usar, y proporciona a los desarrolladores todas las funcionalidades que necesitan para satisfacer los requisitos más complejos a cualquier escala. Se proporcionan drivers para muchos lenguajes, y la comunidad ha desarrollado muchas más.

MongoDB dispone de dos variantes de despliegue:

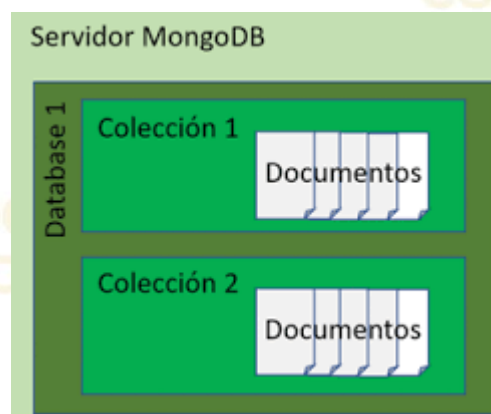
Local: como Mongo Server

Remota: mediante una plataforma configurada, lista en la nube para usar llamada Mongo Atlas

Característica:

- Almacena datos en documentos flexibles similares a JSON: la estructura de datos puede cambiarse con el tiempo.
- El modelo de documento se asigna a los objetos en el código de su aplicación para facilitar el trabajo con los datos.
- Las consultas ad hoc, la indexación y la agregación en tiempo real ofrecen maneras potentes de acceder a los datos y analizarlos.
- MongoDB es una base de datos distribuida en su núcleo, por lo que la alta disponibilidad, la escalabilidad horizontal y la distribución geográfica están integradas y son fáciles de usar.
- MongoDB es de uso gratuito.

Arquitectura Mongo DB



Una colección en Mongo DB es muy similar a una tabla de una base de datos. La tabla

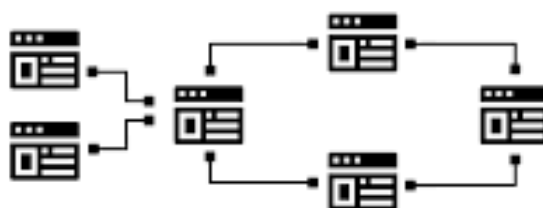
almacena registros (filas) mientras que las colecciones almacenan documentos.

Aquí es donde comienzan las diferencias importantes entre una base de datos relacional y otra base de datos no relacional. El concepto de fila y de documentos son bastante diferentes.

Una fila está compuesta por columnas y siempre son las mismas para todas, mientras que un documento está compuesto por claves y valores, y cada documento puede tener variaciones importantes con respecto al anterior dentro de una colección.

Las limitaciones de las bases de datos relacionales

</> BASE DE DATOS RELACIONAL



1-Dificultades de crecimiento: Cuando estas bases de datos comienzan a crecer en un volumen, el almacenamiento y el costo de mantenimiento se convierte en un problema de alto costo.

2-Cambios en la estructura: el entorno empresarial es altamente dinámico. Esto exige que se realicen cambios de forma eventual en los registros de datos. Si ejecutamos cambios, la base de datos debe ser modificada en su estructura para admitir las modificaciones. Si las modificaciones no se realizan esta se verá afectada y sus procesos interrumpidos.

3-Complejidad en la instalación: Algunas bases de datos SQL se ven condicionadas por el sistema operativo en el cual van a funcionar y los requisitos mínimos de funcionamiento de los servidores u ordenadores.

4-Dificultad de la interfaz: la interfaz de una base de datos SQL es más compleja que

agregar algunas líneas de código.

5-Mas características implementadas de forma patentada: si bien las bases de datosSQL se ajustan a los estándares de calidad, algunas bases de datos implementan extensiones propietarias al SQL estándar para garantizar el bloqueo del proveedor.

Base de datos no relacionales (NOSQL)



En este modelo de base de datos se trabaja con un lenguaje diferente a las relaciones. No tienen un identificador que sirva de relación entre los conjuntos de datos. Estas trabajan bajo formato NoSQL (Not only SQL) en el que los datos no necesariamente se registran en una tabla.

Entre las NoSQL podemos mencionar a Mongo DB, CouchDB, Cassandra y Neo4J. Este tipo de modelos están siendo utilizados en programas y aplicaciones que manejan grandes volúmenes de datos.



CouchDB



cassandra

Este tipo de modelos están siendo utilizados en programas y aplicaciones que manejan grandes volúmenes de datos. Empresas globales como Facebook, Google, Amazon, Twitter entre otras han incorporado con gran éxito las múltiples propiedades y beneficios que nos entregan.

Ventajas de usar Base de datos No Relacionales (NoSQL)

1-Versatilidad: la capacidad de adaptación para brindar soluciones es el punto más fuerte de las NoSQL. Las posibilidades de crecimiento en el volumen de datos o la posibilidad de incluir cambios sobre la forma en que ingresamos los datos sin necesidad de alterar la estructura permiten adaptarse de forma rápida a un entorno de alto dinamismo como en el mundo en que vivimos.

2-Crecimiento horizontal: estas bases de datos son altamente escalables. Es decir, que, si durante requerimos instalar mayor cantidad de nodos para ampliar la capacidad, podemos hacerlo sin problemas. Esto no va a interrumpir la usabilidad o consultas dentro la Base de Datos.

Bajos requerimientos: no necesariamente necesitamos servidores con gran cantidad de recursos para operar. La adaptabilidad y flexibilidad de las bases de datos no relacionales nos permiten empezar con bajos niveles de inversión en equipos e ir ampliando la capacidad a medida que nuestras necesidades.

Modelos de bases de datos NoSQL

Las bases de datos NoSQL utilizan una gran variedad de modelos de datos para acceder y administrar datos. Estos tipos de bases de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas de las restricciones de coherencia de datos en otra base de datos.

Podemos considerar el modelo de base de datos para almacenar libros:

En una base de datos relacional, un registro de un libro a menudo se normaliza y se almacena en tablas separadas, y las relaciones se definen mediante restricciones de claves primarias y externas. El modelo relacional está diseñado para permitir que la base de datos aplique la integridad referencial entre tablas, normalizada para reducir la redundancia, además de estar optimizada para el almacenamiento.

En una base de datos NoSQL, el registro de un libro se almacena generalmente como un documento JSON, en este modelo, los datos están optimizados para un desarrollo intuitivo y escalabilidad horizontal

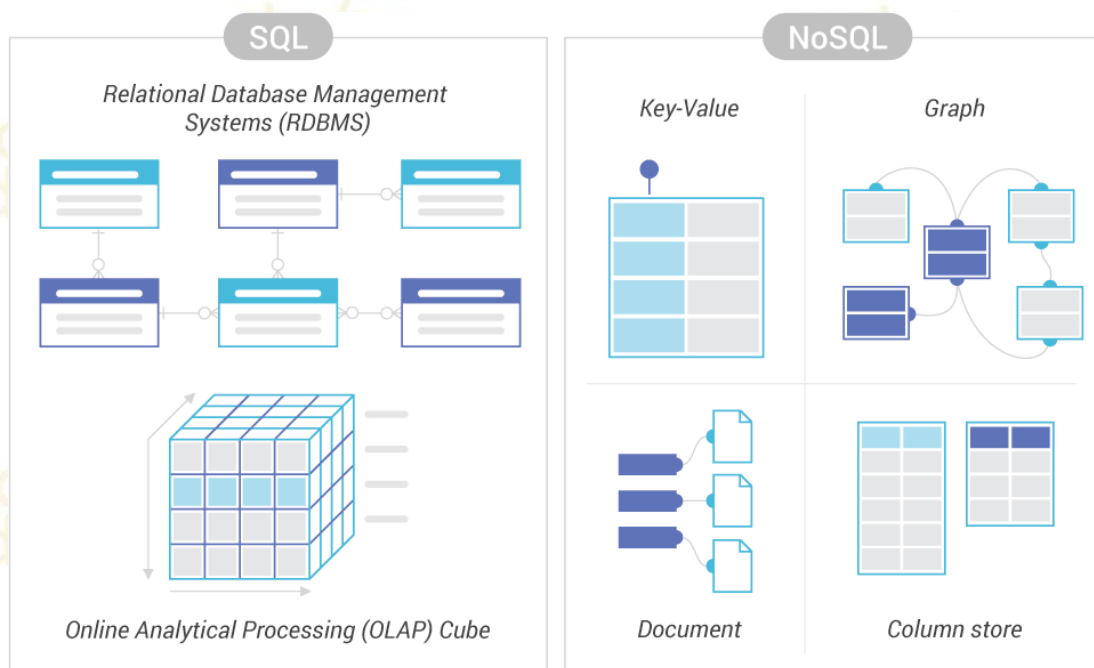
SQL vs NoSQL

Al trabajar con SQL y bases de datos relacionales, el sistema de almacenamiento de datos de nuestras aplicaciones web sigue patrones que otorgan relación entre las tablas de nuestra base de datos con las claves primaria y foráneas.

Esto es altamente beneficioso porque nos genera un sistema de almacenamiento de datos totalmente estructurado.

Al evolucionar nuestras aplicaciones web, debemos realizar modificaciones en nuestro modelo conceptual de la base de datos.

Este problema con las bases de datos NoSQL quedaría resuelto por la flexibilidad que nos ofrecen las bases de datos documentales, ya que estas bases de datos trabajan con datos en formato JSON.



Comparaciones:

MongoDB	MySQL
ALTA DISPONIBILIDAD Ideal si necesitas una alta disponibilidad de los datos con recuperacion automatica, rapida e instantanea	BAJO MANTENIMIENTO Si recién comienzas y no crees que creceras muchos MySQL te ayudara con su facil y bajo mantenimiento.
FRAGMENTACIÓN Si piensas que creceras mucho en el futuro te conviene MongoDB ya que tienen incorporada la fragmentación	PRESUPUESTO LIMITADO Ideal si necesitas alta performance con un presupuesto limitado.
ESQUEMA INESTABLE Ideal si tienes un esquema inestable y cambiante para evitar cambiar el esquema repetidamente	ESQUEMA FIJO Ideal si tu esquema no cambia y siempre guardas los mismos tipos de datos. Por ejemplo un blog.
SIN ADMINISTRADOR DE BD Si no quieres usar un administrador de base de datos. Si creces demasiado deberas utilizar uno de todas maneras.	TRANSACCIONES ALTAS Si requieres de una alta tasa de transacciones
EN LA NUBE Si todos tus servicios estan basados en la nube. MongoDB te resultara mas conveniente.	SEGURIDAD DE LOS DATOS Si la seguridad de los datos es tu prioridad, MySql es el manejador de bases de datos más seguro.

Ventajas Mysql:

- Podemos ejecutar sentencias SQL directamente en nuestra base de datos.
- Posibilidad de abstracción de nuestra base de datos con algún ORM estilo Doctrine o Hibernate.
- Almacenamiento de datos totalmente organizado y jerarquizado con claves primarias y foráneas.
- Nos permite evitar la duplicidad de registros.
- Mejora notable en mantenimiento de datos en relación a la seguridad requerida de los mismos.

Desventajas Mysql:

- Si nuestro sistema escala y evoluciona, tendremos que haber diseñado nuestra base de datos según los posibles nuevos requerimientos.
- Requiere más espacio de almacenamiento que una base NoSQL.
- Las transacciones de datos son más pesadas frente a las bases de datos NoSQL.
- Los límites en los campos de las tablas nos pueden hacer perder datos si no los configuramos adecuadamente según el tamaño del dato que nos puedan introducir los usuarios.

Ventajas NoSQL:

- La escalabilidad y su carácter descentralizado hacen que soporte estructuras distribuidas.
- Permiten realizar sistemas más abiertos y flexibles debido a su fácil adaptación de nuevas evoluciones de nuestras aplicaciones web.
- No se requieren potentes recursos para poder trabajar con bases de datos NoSQL.
- Optimización de las consultas en base de datos para grandes cantidades de datos almacenados.

Desventajas NoSql:

- Problemas con sentencias SQL ya que no admiten el 100% de las consultas existentes.
- No es capaz de realizar transacciones. Si bien en nuestra web o en una aplicación que hemos desarrollado podemos simular una transacción, MongoDB no tiene esa opción entre sus tantas capacidades.
- La principal desventaja de MongoDB es que carece de algo tan fundamental como los Joins.
- Falta de estandarización entre las diferentes bases de datos NoSQL.