



Desarrollar la solución de Software Rap. Construir la Base de Datos

JULIAN ANDRES BOLAÑOS SALINAS

SENA Centro De Comercio Y Servicios- Popayán Cauca
ADSO: 3066446
Juan Pablo Chamizo Vega



Tabla de contenido

Informe: Ventajas Clave de MongoDB y las Bases de Datos No Relacionales.....	4
Aquí se explican sus principales ventajas:	4
1. Flexibilidad en el Esquema	4
2. Escalabilidad Horizontal.....	4
3. Modelado Cercano a Objetos (Documentos JSON).....	4
4. Alta Disponibilidad y Tolerancia a Fallos	4
5. Rapidez en el Manejo de Grandes Volúmenes de Datos	5
6. Soporte para Datos Anidados y Arrays.....	5
Actividad de Reflexión:.....	5
2. ¿En qué casos sería más útil tener documentos con información anidada?	5
CONSULTAS EN MONGODB: Actividad de Transferencia	6
Realizar las siguientes 12 consultas en MongoDB Compass:.....	6
1. Mostrar todos los productos → {}.....	6
2. Buscar productos de la categoría 'Frutas'	6
3. Productos con precio mayor a 3000.....	7
4. Clientes que compraron 'Leche Deslactosada'	7
5. Proyección (mostrar solo nombre y precio).....	8
6. Ordenar productos por precio descendente	8
7. Pedidos con total >= 8000	9
8. Productos cuyo proveedor esté en Bogotá	9
9. Pedidos donde el cliente sea 'Ana López' y contenga más de 1	10
producto.....	10



10. Buscar productos con stock menor a 60	10
11. Contar cuántos productos hay en 'Lácteos'	11
12. Agrupar pedidos por cliente y sumar gasto total.....	11



Informe: Ventajas Clave de MongoDB y las Bases de Datos No Relacionales

Las bases de datos no relacionales, como MongoDB, surgieron para abordar desafíos modernos que las bases de datos tradicionales de tablas fijas tenían dificultades para manejar. Su enfoque flexible y escalable las ha convertido en una pieza fundamental para muchas aplicaciones actuales.

Aquí se explican sus principales ventajas:

1. Flexibilidad en el Esquema

Imagina que estás diseñando una aplicación y necesitas almacenar información de usuarios. Con una base de datos tradicional, deberías definir desde el principio todas las columnas (nombre, email, etc.), y cualquier cambio posterior puede ser complejo. Con MongoDB, es como si tuvieras un cuaderno de notas en blanco para cada usuario. Un usuario puede tener "nombre" y "edad", y otro, sin problema, puede tener "apodo" y "redes sociales". Esta capacidad de adaptarse sobre la marcha, sin interrumpir el servicio, es invaluable para desarrollos ágiles donde los requisitos evolucionan rápidamente.

2. Escalabilidad Horizontal

Piensa en la escalabilidad de dos maneras: vertical (hacer un servidor más grande) y horizontal (añadir más servidores). Las bases no relacionales como MongoDB son expertas en la escalabilidad horizontal. En lugar de depender de una sola máquina súper potente y cara, puedes distribuir la carga de datos y las consultas entre múltiples servidores más simples y económicos. Es como formar un equipo de trabajo en lugar de depender de un solo superhéroe; si la carga aumenta, simplemente añades más miembros al equipo.

3. Modelado Cercano a Objetos (Documentos JSON)

En la programación, los datos se estructuran en objetos. MongoDB almacena la información en documentos en formato JSON (o BSON, su versión binaria), que es un formato extremadamente intuitivo y muy similar a como los desarrolladores piensan y codifican. Esto elimina lo que se conoce como "impedancia de objeto-relacional", que es la fricción y el trabajo extra necesario para convertir los datos de las tablas de una base de datos SQL en objetos utilizables en el código. Con MongoDB, el dato que guardas es casi idéntico al que usas en tu programa.

4. Alta Disponibilidad y Tolerancia a Fallos

La disponibilidad continua es crucial para cualquier aplicación moderna. MongoDB está diseñado para no tener un "punto único de fallo". Por defecto, se configura en réplicas: un servidor principal (primario) y varias copias (secundarias). Si el servidor principal falla por cualquier razón, una de las copias asume el control de forma automática y casi instantánea, garantizando que el servicio siga funcionando sin interrupciones perceptibles para el usuario.



5. Rapidez en el Manejo de Grandes Volúmenes de Datos

Esta ventaja es el resultado directo de las anteriores. La combinación de un modelo de datos simple (documentos), la capacidad de escalar horizontalmente y la ausencia de operaciones complejas como los "JOIN" (que son costosos en SQL), permite que MongoDB maneje operaciones de lectura y escritura a muy alta velocidad, incluso con terabytes de información. Está optimizado para rendir bien en escenarios donde el volumen, la velocidad y la variedad de los datos son grandes.

6. Soporte para Datos Anidados y Arrays

En la vida real, la información no siempre es plana. Un usuario puede tener múltiples direcciones, o una orden de compra puede contener una lista de productos. MongoDB maneja esto de forma nativa. Puedes tener documentos con objetos dentro de otros objetos y con listas (arrays) de valores, todo dentro de un mismo registro. Esto permite representar estructuras de datos complejas de una manera muy natural y eficiente, sin necesidad de crear múltiples tablas relacionadas y unirlos después con consultas complejas

Actividad de Reflexión:

1. ¿Por qué crees que algunas empresas prefieren usar bases NoSQL en lugar de bases SQL? Sí, es acertado afirmar que, para muchos casos de uso modernos, las bases de datos no relacionales (NoSQL) presentan ventajas decisivas. La clave reside en que la menor rigurosidad en la estructura y la consistencia inmediata se traduce directamente en mayor flexibilidad, velocidad y escalabilidad, factores críticos para las empresas hoy en día.

2. ¿En qué casos sería más útil tener documentos con información anidada?

Efectivamente, la estructura de documentos con información anidada (o embebida) es más útil en casos donde los datos tienen una relación de "pertenencia" o una conexión fuerte y directa con la entidad principal. Este modelo, típico de bases de datos NoSQL de documentos como MongoDB, es ideal cuando la información anidada se accede casi siempre junto con la principal.

3. ¿Cómo se relaciona MongoDB Compass con la facilidad de uso frente a la consola?

MongoDB Compass se relaciona directamente con la facilidad de uso al ofrecer una interfaz gráfica intuitiva (GUI) que abstrae la complejidad de los comandos de la consola, haciendo la gestión de la base de datos más accesible y visual, especialmente para usuarios menos técnicos o durante las etapas de desarrollo y prototipado.



CONSULTAS EN MONGODB:

Actividad de Transferencia

Realizar las siguientes 12 consultas en MongoDB Compass:

1. Mostrar todos los productos → {}

The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' list with 'localhost:27017' selected. The main panel shows the 'Productos' collection with 10 documents. The query field is empty, and the 'Find' button is highlighted. The table below represents the data shown in the interface:

_id	ObjectId	nombre	String	precio	Int32	categoria	String	stock	Int32
1	ObjectId('68d5a4fe37ffe4...')	"Manzanas Orgánicas"		3500		"frutas"		150	
2	ObjectId('68d5a4fe37ffe4...')	"Bananos Premium"		2800		"frutas"		45	
3	ObjectId('68d5a4fe37ffe4...')	"Naranjas Valencia"		3200		"frutas"		80	
4	ObjectId('68d5a4fe37ffe4...')	"Leche Entera Orgánica"		4500		"lacteos"		75	
5	ObjectId('68d5a4fe37ffe4...')	"Leche Deslactosada"		5200		"lacteos"		40	
6	ObjectId('68d5a4fe37ffe4...')	"Yogurt Natural"		3800		"lacteos"		55	
7	ObjectId('68d5a4fe37ffe4...')	"Queso Campestino"		8500		"lacteos"		25	
8	ObjectId('68d5a4fe37ffe4...')	"Zanahorias Baby"		2500		"verduras"		120	
9	ObjectId('68d5a4fe37ffe4...')	"Lechuga Hidropónica"		1800		"verduras"		35	
10	ObjectId('68d5a4fe37ffe4...')	"Tomates Cherry"		4200		"verduras"		60	

2. Buscar productos de la categoría 'Frutas'

The screenshot shows the MongoDB Compass interface with a query filter applied: `{ categoria: "frutas" }`. The 'Find' button is highlighted, and the results show 3 documents. The table below represents the data shown in the interface:

_id	ObjectId	nombre	String	precio	Int32	categoria	String	stock	Int32
1	ObjectId('68d5a4fe37ffe4...')	"Manzanas Orgánicas"		3500		"frutas"		150	
2	ObjectId('68d5a4fe37ffe4...')	"Bananos Premium"		2800		"frutas"		45	
3	ObjectId('68d5a4fe37ffe4...')	"Naranjas Valencia"		3200		"frutas"		80	



3. Productos con precio mayor a 3000

MongoDB Compass interface showing the 'Productos' collection in the 'EcoMarket' database. The query filter is `{ "precio": { "$gt": 3000 } }`. The results show 7 documents.

_id	ObjectId	nombre	precio	categoria	stock
1	ObjectId('68d5a4fe37ffe4...')	"Manzanas Orgánicas"	3500	"frutas"	150
2	ObjectId('68d5a4fe37ffe4...')	"Naranjas Valencia"	3200	"frutas"	80
3	ObjectId('68d5a4fe37ffe4...')	"Leche Entera Orgánica"	4500	"lacteos"	75
4	ObjectId('68d5a4fe37ffe4...')	"Leche Deslactosada"	5200	"lacteos"	40
5	ObjectId('68d5a4fe37ffe4...')	"Yogurt Natural"	3800	"lacteos"	55
6	ObjectId('68d5a4fe37ffe4...')	"Queso Campestino"	8500	"lacteos"	25
7	ObjectId('68d5a4fe37ffe4...')	"Tomates Cherry"	4200	"verduras"	60

4. Clientes que compraron 'Leche Deslactosada'

MongoDB Compass interface showing the 'Pedidos' collection in the 'EcoMarket' database. The query filter is `{ "productos": { "$elemMatch": { "nombre": "Leche Deslactosada" } } }`. The results show 2 documents.

_id	ObjectId	numero_pedido	cliente	fecha_pedido	productos
1	ObjectId('68d5a4e837ffe4...')	"ECO-2024-001"	{ } 4 fields	"2024-11-15"	[] 2 elem
2	ObjectId('68d5a4e837ffe4...')	"ECO-2024-004"	{ } 4 fields	"2024-11-18"	[] 1 elem



5. Proyección (mostrar solo nombre y precio)

Compass interface showing the projection of documents from the 'Productos' collection. The query is set to show only the 'nombre' and 'precio' fields.

Query: `{ "nombre": "Manzanas Orgánicas", "precio": 3500, }`

Results:

_id	nombre	precio	categoria	stock
Object('68d5a4fe37ffe4...')	Manzanas Orgánicas	3500	frutas	150

6. Ordenar productos por precio descendente

Compass interface showing the sorting of documents from the 'Productos' collection by price in descending order.

Query: `{ }`

Sort: `{ "precio": -1 }`

Collation: `{ locale: 'simple' }`

Index Hint: `{ field: -1 }`

Results:

_id	nombre	precio	categoria	stock	descripcion	fecha_vencimiento	proveedor
Object('68d5a4fe37ffe492034110ba')	Leche Deslactosada	5200	lacteos	40	Leche deslactosada para personas con intolerancia a la lactosa, 100% n.m.	2024-11-28	Object



7. Pedidos con total >= 8000

MongoDB Compass - localhost:27017/EcoMarket.Pedidos

connections Edit View Collection Help

Compass

My Queries

CONNECTIONS (1)

Search connections

localhost:27017

- EcoMarket
 - Pedidos
 - Productos
 - Tienda
 - admin
 - config
 - local

localhost:27017 > EcoMarket > Pedidos

Documents 6 Aggregations Schema Indexes 1 Validation

Generate query Explain Reset Find Options

ADD DATA EXPORT DATA UPDATE DELETE

25 1 - 4 of 4

```
{ "total": { "$gte": 8000 } }
```

```
{
  "_id": ObjectId('68d5a4e837ffe492034110ae'),
  "numero_pedido": "ECO-2024-001",
  "cliente": Object,
  "fecha_pedido": "2024-11-15",
  "productos": Array (2),
  "total": 13900,
  "estado": "entregado",
  "fecha_entrega": "2024-11-16"
}
```

```
{
  "_id": ObjectId('68d5a4e837ffe492034110af'),
  "numero_pedido": "ECO-2024-002",
  "cliente": Object,
  "fecha_pedido": "2024-11-16",
  "productos": Array (1),
  "total": 8400,
  "estado": "en_proceso",
  "fecha_entrega": null
}
```

```
{
  "_id": ObjectId('68d5a4e837ffe492034110b0'),
  "numero_pedido": "ECO-2024-003"
}
```

8. Productos cuyo proveedor esté en Bogotá

MongoDB Compass - localhost:27017/EcoMarket.Productos

connections Edit View Collection Help

Compass

My Queries

CONNECTIONS (1)

Search connections

localhost:27017

- EcoMarket
 - Pedidos
 - Productos
 - Tienda
 - admin
 - config
 - local

localhost:27017 > EcoMarket > Productos

Documents 10 Aggregations Schema Indexes 1 Validation

Generate query Explain Reset Find Options

ADD DATA EXPORT DATA UPDATE DELETE

25 1 - 5 of 5

```
{ "proveedor.ciudad": "Bogotá" }
```

```
{
  "precio": 2800,
  "categoria": "frutas",
  "stock": 45,
  "descripcion": "Bananos orgánicos maduros y dulces de primera calidad",
  "fecha_vencimiento": "2024-11-30",
  "tipo_cultivo": "Orgánico",
  "proveedor": {
    "nombre": "Frutas del Trópico",
    "contacto": {
      "ciudad": "Bogotá"
    }
  }
}
```

```
{
  "_id": ObjectId('68d5a4fe37ffe492034110ba'),
  "nombre": "Leche Deslactosada",
  "precio": 5200,
  "categoria": "Lacteos",
  "stock": 40,
  "descripcion": "Leche deslactosada para personas con intolerancia a la lactosa, 100% n.",
  "fecha_vencimiento": "2024-11-28",
  "tipo": "Deslactosada",
  "proveedor": {
    "nombre": "Lácteos Especiales Bogotá",
    "contacto": {
      "ciudad": "Bogotá"
    }
  }
}
```



9. Pedidos donde el cliente sea 'Ana López' y contenga más de 1 producto.

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure: **localhost:27017** > **EcoMarket** > **Pedidos**. The main panel shows the **Pedidos** collection with 6 documents. A query is entered in the **Documents** tab: `{ "cliente.nombre": "Ana López", "productos.2": { "$exists": true } }`. The query results show a document for a client named 'Ana López' with two products: 'Bananas Premium' and 'Leche Deslactosada'.

10. Buscar productos con stock menor a 60

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure: **localhost:27017** > **EcoMarket** > **Productos**. The main panel shows the **Productos** collection with 10 documents. A query is entered in the **Documents** tab: `{ "stock": { "$lt": 60 } }`. The query results show two documents: 'Bananas Premium' with a stock of 45 and 'Leche Deslactosada' with a stock of 40.



11. Contar cuántos productos hay en 'Lácteos'

MongoDB Compass - localhost:27017/EcoMarket/Productos

Connections Edit View Collection Help

Compass

My Queries

CONNECTIONS (1)

Search connections

localhost:27017

EcoMarket

Pedidos

Productos

Tienda

admin

config

local

localhost:27017 > EcoMarket > Productos

Documents 10 Aggregations Schema Indexes 1 Validation

Generate query Explain Reset Find Options

Project { field: 0 }

Sort { field: -1 } or [['field', -1]]

Collation { locale: 'simple' }

Index Hint { field: -1 }

Max Time MS 60000

Skip 2 Limit 2

ADD DATA EXPORT DATA UPDATE DELETE

25 1 - 2 of 2

`{ "categoria": "Lacteos" }`

`{ "_id": ObjectId('68d5a4fe37ffe492834110bb'), nombre: "Yogurt Natural", precio: 3800, categoria: "Lacteos", stock: 55, descripcion: "Yogurt natural sin azúcar añadida, con probióticos naturales", fecha_vencimiento: "2024-11-22", proveedor: Object }`

`{ "_id": ObjectId('68d5a4fe37ffe492834110bc'), nombre: "Queso Campesino", precio: 8500, categoria: "Lacteos" }`

12. Agrupar pedidos por cliente y sumar gasto total

Connections Edit View Collection Help

Compass

My Queries

CONNECTIONS (1)

Search connections

localhost:27017

EcoMarket

Pedidos

Productos

Tienda

admin

config

local

localhost:27017 > EcoMarket > Pedidos

Documents 6 Aggregations Schema Indexes 1 Validation

Open MongoDB shell

\$group Edit Explain Export Run Options

ALL RESULTS Showing 1 - 5 count results

`{ "_id": "Ana López", gasto_total: 34200, cantidad_pedidos: 2 }`

`{ "_id": "Carlos Mendoza", gasto_total: 8400, cantidad_pedidos: 1 }`

`{ "_id": "María Rodríguez", gasto_total: 5200, cantidad_pedidos: 1 }`

`{ "_id": "Pedro Sánchez", gasto_total: 8900, cantidad_pedidos: 1 }`

`{ "_id": "Laura García" }`

