



# Programa de formación MACHINE LEARNING AND DATA SCIENCE MLDS

Facultad de  
**INGENIERÍA**



# Módulo 3

## Big Data

Unidad 3

Bases NoSQL documentales

Clase sincrónica

Facultad de  
**INGENIERÍA**





## Bienvenida

# Jorge Eliécer Camargo Mendoza, PhD.

<https://dis.unal.edu.co/~jecamargom/>

[jecamargom@unal.edu.co](mailto:jecamargom@unal.edu.co)



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

Departamento de Ingeniería de Sistemas e Industrial

Facultad de Ingeniería

Universidad Nacional de Colombia

Sede Bogotá



## Tabla de contenidos

- 1 ¿Qué son las bases documentales?
- 2 Ejemplos bases documentales
- 3 ¿Qué es un documento?
- 4 ¿Qué es MongoDB?
- 5 ¿Quién usa MongoDB?
- 6 Arquitectura de MongoDB
- 7 Modelo de datos
- 8 Conceptos de MongoDB (CRUD)
- 9 ¿Qué son los shards?
- 10 Map – Reduce

## Objetivos de aprendizaje

## Unidad 3 – Bases NoSQL Documentales

**Al finalizar la unidad usted deberá ser capaz de:**

1



Entender el modelo de datos y la arquitectura en bases de datos documentales como MongoDB.

2



Utilizar el lenguaje de consulta de MongoDB para manipular información desde el lenguaje de programación Python.

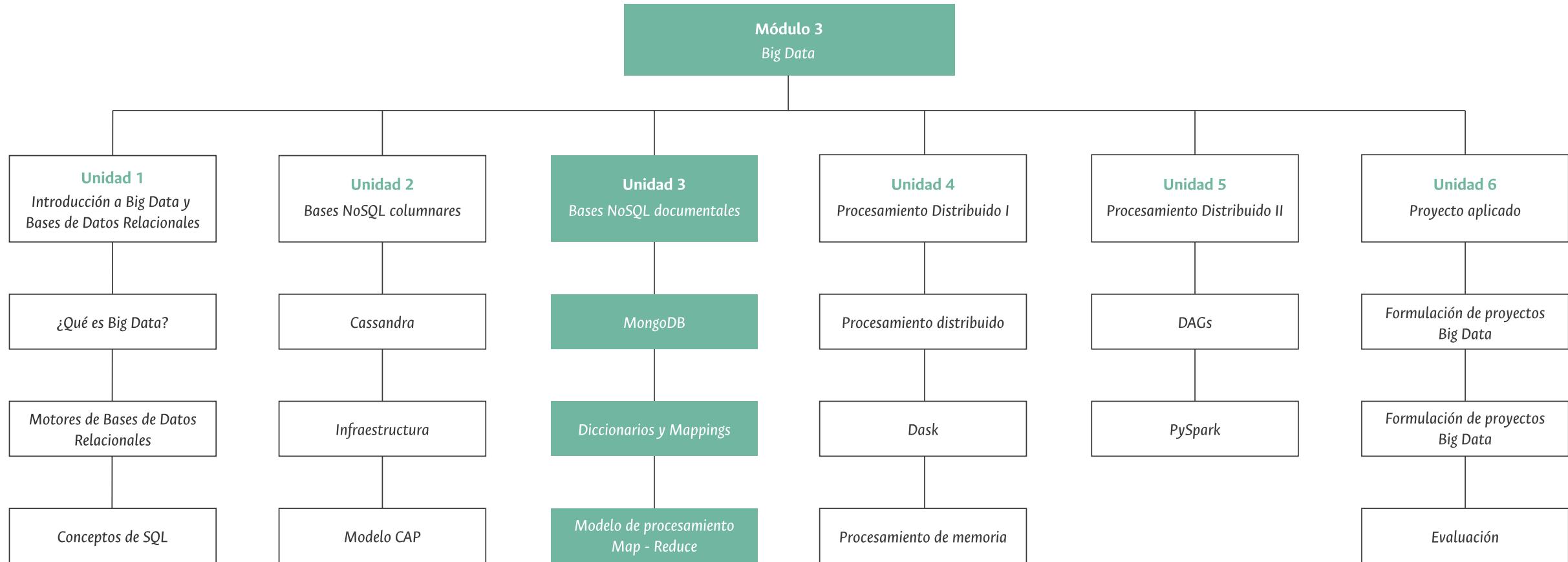
3



Entender los conceptos de computación distribuida con operaciones de tipo Map - Reduce desde MongoDB.



# Mapa de contenidos de la unidad



## 1

# ¿Qué son las bases de datos documentales?

Las bases de datos documentales constituyen una de las principales subcategorías dentro de las denominadas bases de datos NoSQL que combinan la ausencia de un esquema con la posibilidad de estructurar los datos almacenados.

Las bases de datos documentales están diseñadas para almacenar datos semiestructurados, como documentos, normalmente en formato JSON o XML. A diferencia de las bases de datos relacionales tradicionales, el modelo de datos en una base de datos de documentos no está estructurado en un formato de tabla de filas y columnas.



```
{  
  "id": 1,  
  "nombre": "Bolt",  
  "dueno": "Jhon Gonzalez",  
  "especie": "Perro",  
  "fecha_nacimiento": "2017-02-23"  
}
```



```
{  
  "id": 2,  
  "nombre": "Luna",  
  "dueno": "Camila Martinez",  
  "fecha_nacimiento": "2020-12-12",  
  "especie": "Perro",  
  "raza": "Dálmata"  
}
```

## 2 Ejemplos de bases de datos documentales

### eXist - db

Sistema de gestión de bases de datos libre y de código abierto. Utiliza principalmente documentos XML.



### MongoDB

Es de código abierto, almacena los datos en documentos flexibles



mongoDB

### CouchDB

Destaca por su facilidad de uso y comprensión. Utiliza el formato JSON.



CouchDB  
relax

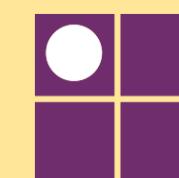
### Firestore

Alojada en la nube a la que pueden acceder apps iOS, Android y Web.



### BaseX

Especializado en almacenar, consultar y visualizar colecciones y documentos XML de gran tamaño.



### SimpleDB

Desarrollado por Amazon, está diseñado para almacenar tuplas de información.



### CosmosDB

Administra el crecimiento de los datos dividiendo horizontalmente la carga de trabajo.

## 3

## ¿Qué es un documento?

Es un **registro** en una base de datos NoSQL orientada a documentos que se encarga de almacenar la información sobre un objeto y cualquiera de sus metadatos relacionados. Cada documento funciona como una unidad autónoma de información. Los documentos almacenan datos en pares **llave - valor** o **key - value**. El valor es usualmente un dato estructurado con una variedad de tipos aceptados, incluidas cadenas, números, fechas, matrices y objetos.



```
lib_1
{
  "id": 1,
  "autor": "Gabriel García Márquez",
  "título": "100 años de soledad",
  "anio_publicacion": "1967",
  "unidades": 1258
}
```



```
lib_2
{
  "id": 2,
  "autor": "Julio Cortázar",
  "título": "Rayuela",
  "anio_publicacion": "1963",
  "unidades": 526,
  "genero": "Novela"
}
```



Cada documento tiene un identificador único (señalado en color naranja). Generalmente esta clave se compone de una cadena pero, en algunos casos, puede tratarse de una URI que sirve para rescatar el documento de la base de datos.

Al contrario que una base de datos relacional en la que todos los registros deben tener los mismos atributos, los cuales pueden quedar vacíos, en un documento no quedan 'campos' vacíos, nótese que en *lib\_1* no aparece el campo “género” mientras que en *lib\_2* si está presente. De este modo es posible añadir nueva información sin necesidad de establecer qué información queda excluida.

## ¿Qué es un documento?

## Formatos de documento



### JSON - JavaScript Object Notation

Es un formato de intercambio de datos inteligible por los programadores y sencillo de interpretar y crear para las máquinas. Es un formato independiente de los lenguajes de programación, es decir, el servicio que lo consume puede estar desarrollado en un lenguaje diferente al que lo emite e, igualmente, realizar las operaciones necesarias para responder. Soporta objetos embebidos.



### BSON – Binary JSON

Es una representación binaria de las estructuras de datos. Está diseñado para tener mayor almacenamiento y una velocidad más eficiente que los JSONs ya que codifica el tipo y el tamaño de la información. BSON admite la incorporación de objetos embebidos igual que el JSON.



### XML - Extensible Markup Language

Es un estándar abierto, flexible y ampliamente utilizado para almacenar, publicar e intercambiar cualquier tipo de información. Utiliza una estructura de etiquetas para representar elementos de datos.

## ¿Qué es un documento?

## Tipos de datos básicos de un JSON

- **null**: se puede utilizar para representar tanto un valor nulo como un campo inexistente

```
{  
  "x": null  
}
```

- **boolean**: se utiliza para los valores 'true' o 'false'.

```
{  
  "x": true  
}
```

- **integer**: este tipo se utiliza para almacenar un valor numérico. No se deben colocar comillas.

```
{  
  "x": 8  
}
```

- **double**: se utiliza para almacenar valores de punto flotante.

```
{  
  "x": 9.8  
}
```

- **array**: almacena listas de elementos.

```
{  
  "x": ["Coldplay", "Bon Jovi", "Metallica"]  
}
```

- **string**: Se puede representar cualquier cadena de caracteres UTF-8.

```
{  
  "x": "Universidad Nacional"  
}
```

## ¿Qué es un documento?

## Tipos de datos básicos de un JSON

- **Documento embebido (Object):** Un documento embebido es aquel que su contenido se almacena completamente dentro de otro documento.

```
{  
    "cancion": "Let It Be",  
    "album": {  
        "nombre": "Let It Be",  
        "fecha": "8 de mayo de 1970",  
        "genero": "Rock",  
        "num_canciones": 12  
    },  
    "artista": {  
        "nombre": "The Beatles",  
        "origen": "Liverpool",  
        "banda": 'true'  
    },  
    "duracion": "3:50"  
}
```

- **object id:** contiene la identificación de objeto la cual es única y tiene un tamaño de 12 bytes para documentos.

```
{  
    "x": ObjectId()  
}
```

- El JSON Object comienza y termina con llaves {}.
- Las claves deben ser únicas y estar entre comillas dobles.

## 4

## ¿Qué es MongoDB?

MongoDB es una base de datos NoSQL, distribuida, de código abierto, de propósito general orientada a documentos.

MongoDB almacena los datos en documentos flexibles, en formato **BSON**, la representación binaria de JSON, con un esquema dinámico, por lo tanto, los campos pueden variar entre documentos y las estructuras de datos pueden ser modificadas posteriormente.

En relación con el teorema CAP, MongoDB es un almacén de datos de CP, aborda el **particionamiento** de red manteniendo y la **consistencia**.



**mongoDB**

## MongoDB

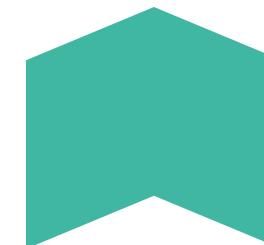
### Ventajas

#### 01 Escalabilidad



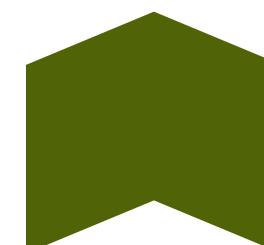
MongoDB, al ser una base de datos distribuida, es fácilmente escalable de forma vertical y horizontal. Mongo tiene la capacidad de crear nuevos nodos, esto permite que el rendimiento sea mucho mejor para la aplicación. También funciona en máquinas de pocos recursos por lo que no habrá limitaciones físicas del hardware en el que se ejecute el servidor.

#### 02 Balanceo de carga



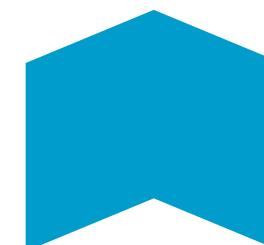
MongoDB tiene la capacidad de ejecutarse de manera simultánea en múltiples servidores, ofreciendo un balanceo de carga y servicio de replicación de datos, de modo que podemos mantener el sistema funcionando en caso de un fallo del hardware.

#### 03 Indexación

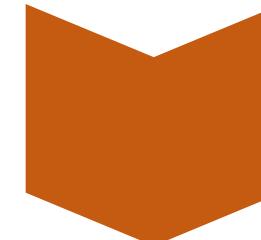


MongoDB tiene gran facilidad de indexación porque estos se pueden administrar y rehacer sin grandes dificultades. Todos los documentos se indexan automáticamente con la llave `_id`. Esto se considera un caso especial porque no se puede eliminar este índice; es lo que asegura que cada valor sea único.

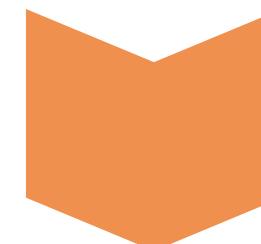
#### 04 Replicación



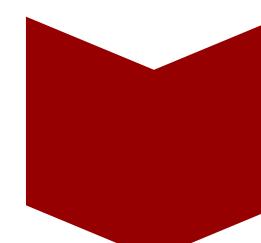
MongoDB proporciona **redundancia** con la ayuda de la replicación. Crea múltiples copias de los datos y envía estas copias a un servidor diferente. Tiene una replicación incorporada con auto - elecciones, es decir, con MongoDB podrás configurar una base de datos secundaria que se puede convertir de manera automática en una base de datos principal si la base de datos principal (initial) deja de estar disponible.

**MongoDB****Desventajas****01** **Transacciones**

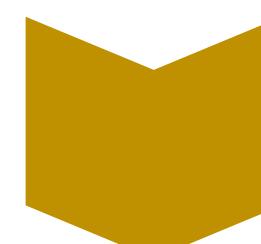
No es una solución adecuada para aplicaciones con transacciones complejas.

**02** **Joins**

Otra característica que tiene esta base de datos, es que no tiene JOINS o uniones para la realización de consultas, por lo que no es posible combinar o relacionar diferentes tablas en un mismo query.

**03** **Tamaño**

No se pueden tener documentos en MongoDB de más de 4 MB.

**04** **Duplicación**

MongoDB, por no dar soporte a JOINS y al no poseer relaciones muy bien definidas entre los documentos o colecciones, conlleva a la duplicación de datos y una redundancia de los mismos provocando un aumento en el uso de la memoria.

5

## ¿Quién usa MongoDB?

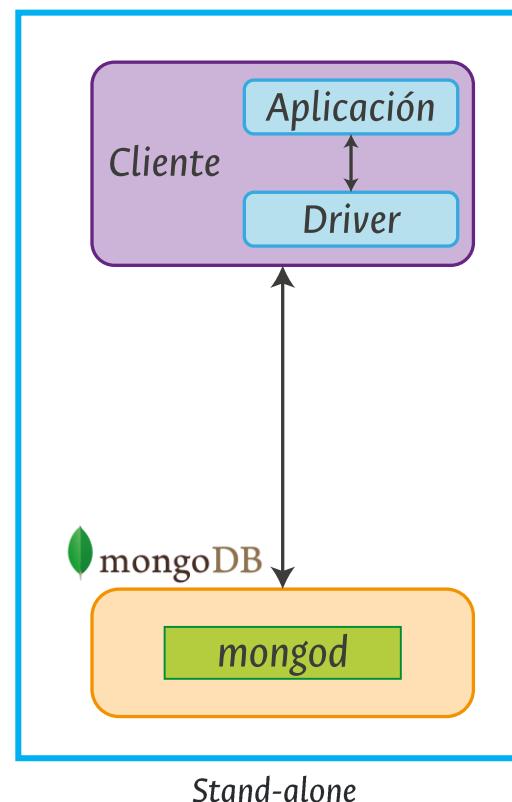
	MongoDB ahora funciona como el principal almacenamiento de datos en tiempo real para las interacciones sociales de WebEx. Gracias a Map – Reduce obtiene y procesa datos de estadísticas.
	Utilizan MongoDB para almacenar los datos del vehículo para poder hacer diagnósticos de necesidad de mantenimiento preventivo.
	Construyó todo su sistema de gestión de contenidos en MongoDB. Además, utiliza MongoDB para analítica en tiempo real, al igual que el <b>Ayuntamiento de Chicago</b> para convertirse en ciudad inteligente.
	La Organización Europea para la Investigación Nuclear utiliza MongoDB para manejar los grandes volúmenes de datos que genera el acelerador de partículas y los resultados de todos los experimentos que desarrollan.
	MongoDB les proporciona una distribución masiva de datos tanto para lectura como para la resiliencia de los mismos gracias a su escalabilidad.



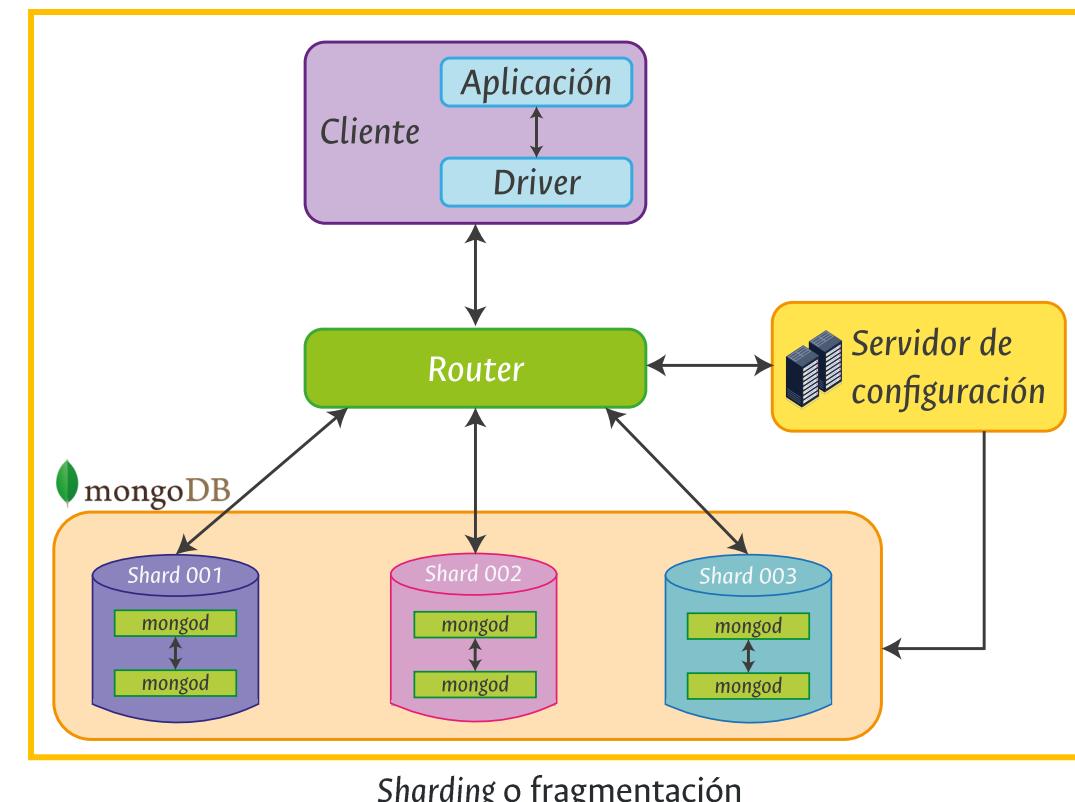
## 6

# Arquitectura de MongoDB

MongoDB ofrece dos tipos de conexión a las bases de datos: la conexión **stand-alone**, lugar en el que el cliente se conecta directamente con el nodo en el cual se ejecuta MongoDB, y una conexión en la que se utiliza el **sharding o fragmentación**, en donde el cliente se comunica con un nodo que ejecuta un componente el cual cumple la función de enrutador y gestiona las peticiones para dirigirlas a los shards o fragmentos correspondientes.



Stand-alone

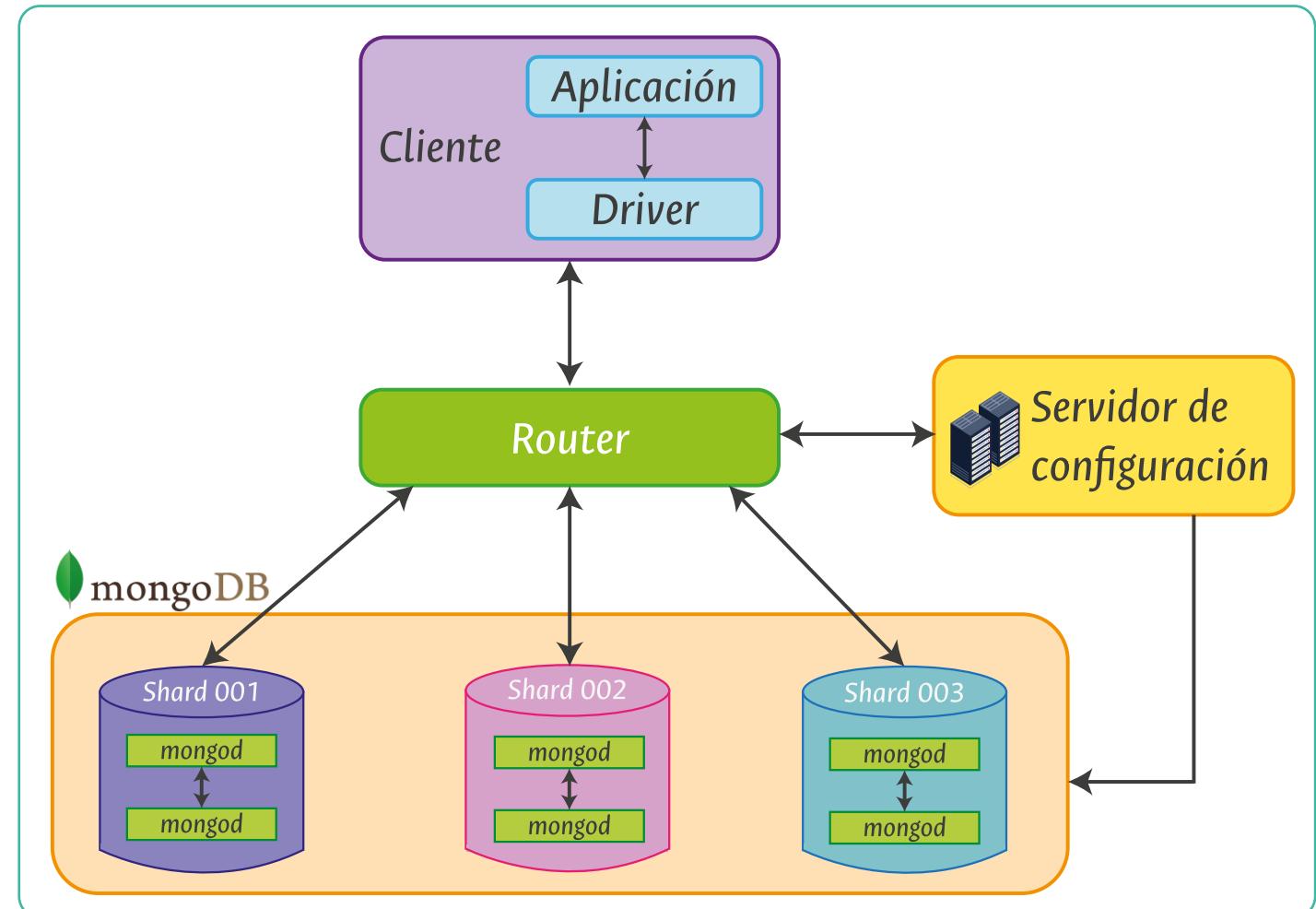


Sharding o fragmentación

## Arquitectura de MongoDB

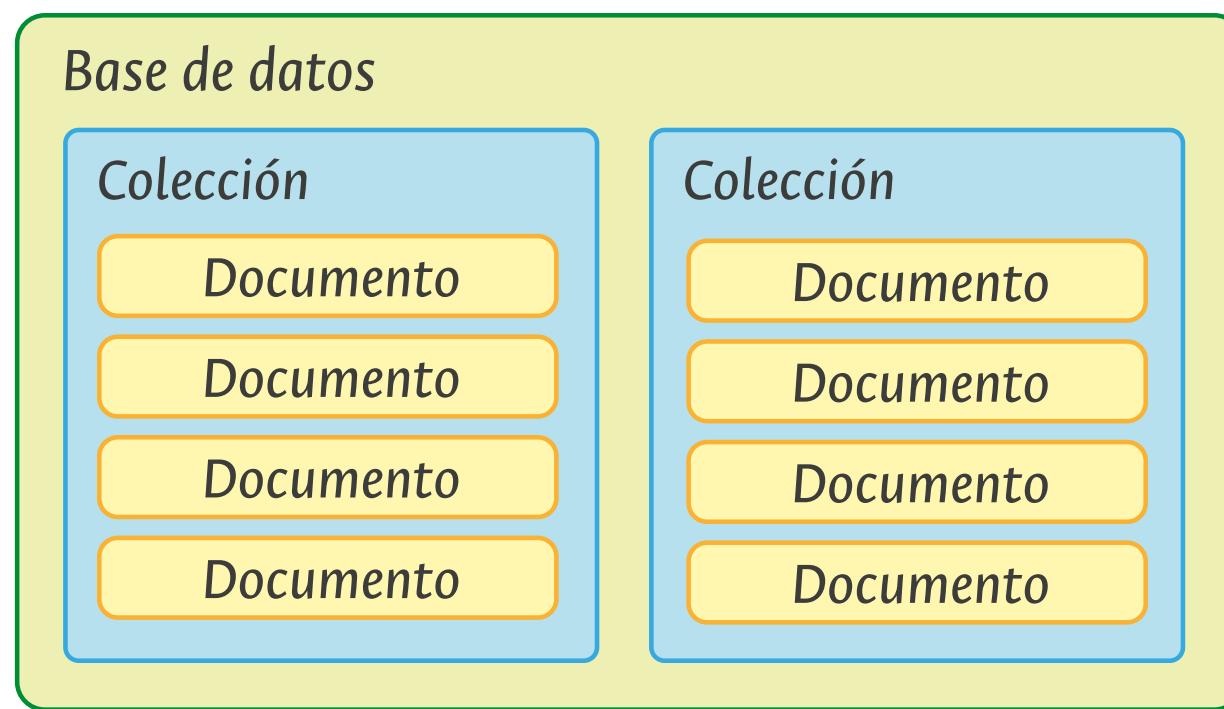
## Componentes de la arquitectura Sharding

- Router: Es la única instancia que puede comunicarse fuera del clúster. Solo es necesaria cuando se escala horizontalmente.
- Servidor de configuración: actúa como el contenedor de los metadatos sobre los documentos almacenados en mongod. Es el encargado de recuperar el sistema en caso de que algún nodo falle. En un clúster puede haber 1 o 3 instancias de este servidor.
- Shard: son conocidos como los set de réplicas ya que contienen a un conjunto de mongod.
- mongod: nodo que se utiliza para almacenar y recuperar los datos.



## Arquitectura de MongoDB

### Componentes de un mongod



#### Base de datos

Sirve como contenedor para las colecciones. Se pueden crear varias bases de datos en una instancia mongod.

#### Colecciones

Una colección en MongoDB es muy similar a una Tabla de una base de datos. La tabla almacena registros (filas) mientras que las colecciones almacenan documentos, sin embargo, no imponen un esquema estricto.

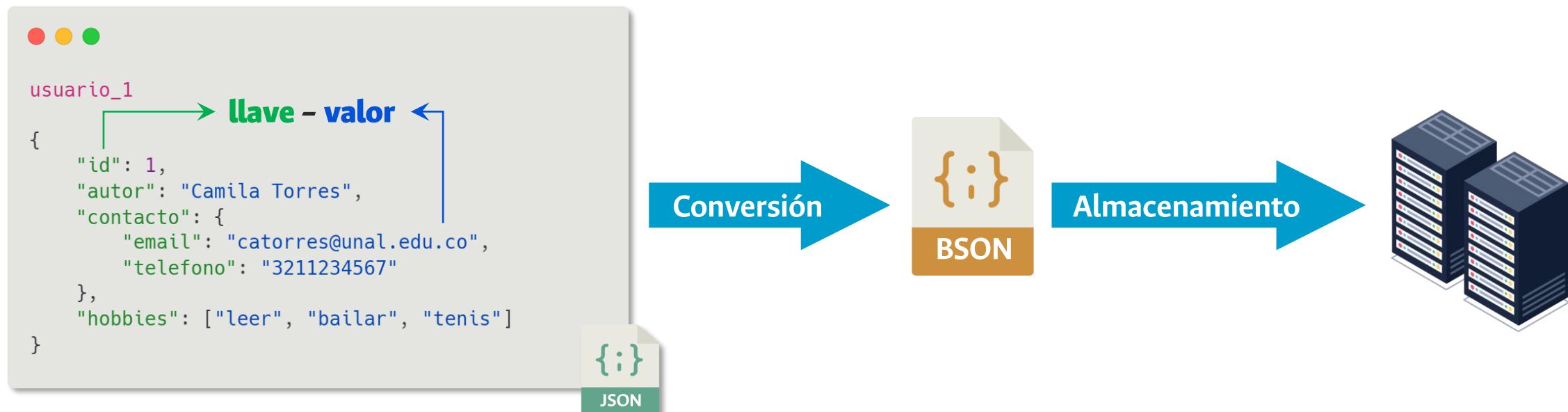
#### Documento

Unidad básica de MongoDB, está compuesto por llaves y valores (*key – value*) y cada documento puede tener variaciones con respecto a los otros de su misma colección.

## 7

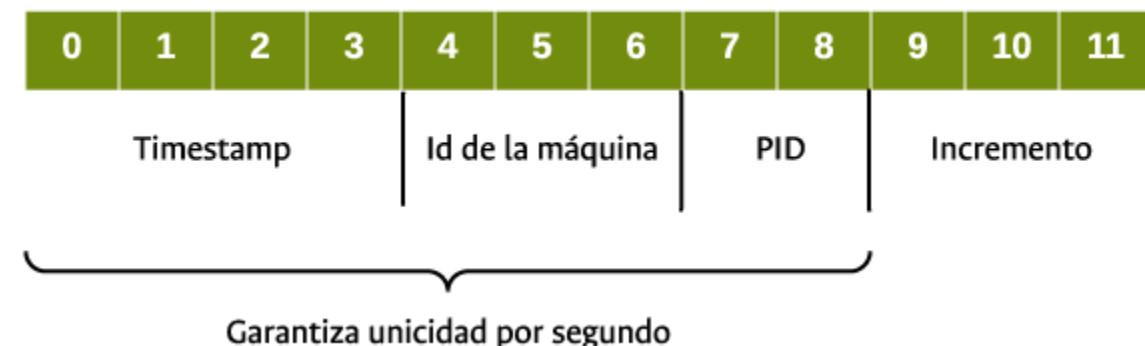
## Modelo de datos

El formato de los documentos **almacenados** en MongoDB es **BSON** (Binary JSON, JavaScript Object Notation); estos documentos están compuestos por parejas **llave - valor** o **key - value**. Este formato hace que MongoDB sea aún más rápido al facilitar que un máquina procese y busque documentos.



**MongoDB** **Modelo de datos**

Estos documentos cuentan con un campo especial de 12 bytes que les garantiza la **identificación única** dentro de las colecciones, este es el **ObjectId**, el cual es generado utilizando un timestamp, el identificador único de la máquina, un PID (identificador de proceso) y un contador local que incrementa secuencialmente.



# Conceptos de MongoDB (CRUD)

## CREATE



Se utiliza para insertar nuevos documentos



```
db.MascotasDB.insert_one({  
  "nombre": "Alana",  
  "especie": "Perro"  
})
```

## READ



Se utiliza para realizar búsquedas y consultas.



```
db.MascotasDB.find()  
> {"_id": ObjectId("1"), "nombre":  
  "Alana", "especie": "Perro"}  
> {"_id": ObjectId("2"), "nombre":  
  "Bolt", "especie": "Perro"}  
> {"_id": ObjectId("3"), "nombre":  
  "Mota", "especie": "Gato"}
```

## UPDATE



Se utiliza para actualizar los documentos.



```
db.MascotasDB.update_one(  
  {_id: ObjectId(1)},  
  {"$set": {"especie": "Gato"}},  
  {"acknowledged": true,  
   "matchedCount": 1, "modifiedCount": 1}  
  
db.MascotasDB.find({"nombre": "Alana"})  
> {"_id": ObjectId("1"), "nombre":  
  "Alana", "especie": "Gato"}  
> {"_id": ObjectId("3"), "nombre":  
  "Mota", "especie": "Gato"}
```

## DELETE



Se utiliza para eliminar los documentos.



```
db.MascotasDB.delete_one(  
  {"nombre": "Bolt"},  
  {"acknowledged": true,  
   "deletedCount": 1}  
  
db.MascotasDB.find()  
> {"_id": ObjectId("1"), "nombre":  
  "Alana", "especie": "Gato"}  
> {"_id": ObjectId("3"), "nombre":  
  "Mota", "especie": "Gato"}
```

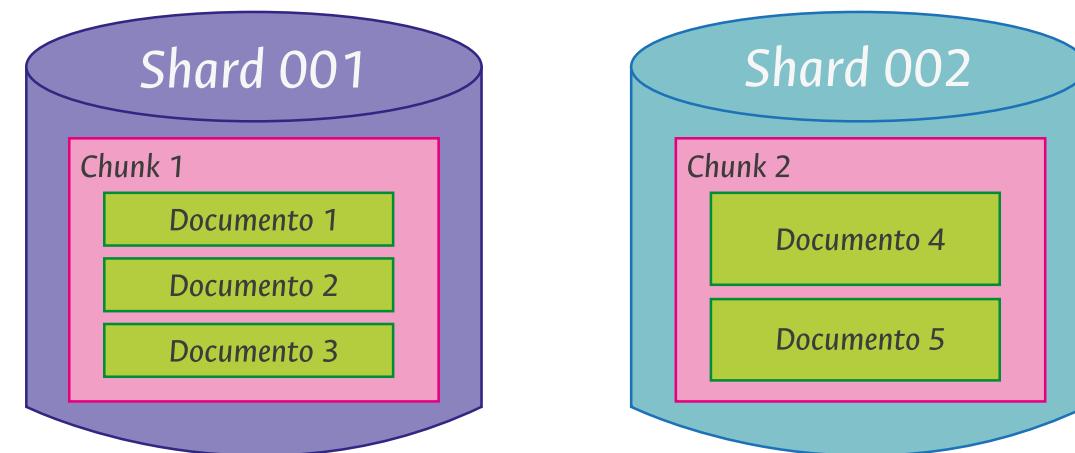
9

## ¿Qué son los shards?

Cuando no es posible almacenar la totalidad de datos en un único servidor, MongoDB permite una operación de **escalado horizontal**. Esto quiere decir que la información puede repartirse en varios servidores, de forma que cada servidor tenga una parte del conjunto completo de datos. A este proceso se le llama **sharding** o fragmentación. El conjunto de servidores que contiene la totalidad de un conjunto de datos se le denomina **sharded cluster** o clúster fragmentado.



El factor de replicación de un sistema está determinado por la cantidad de nodos de datos en un **shard** o fragmento.



Cada uno de los fragmentos resultantes es conocido como un **shard**.

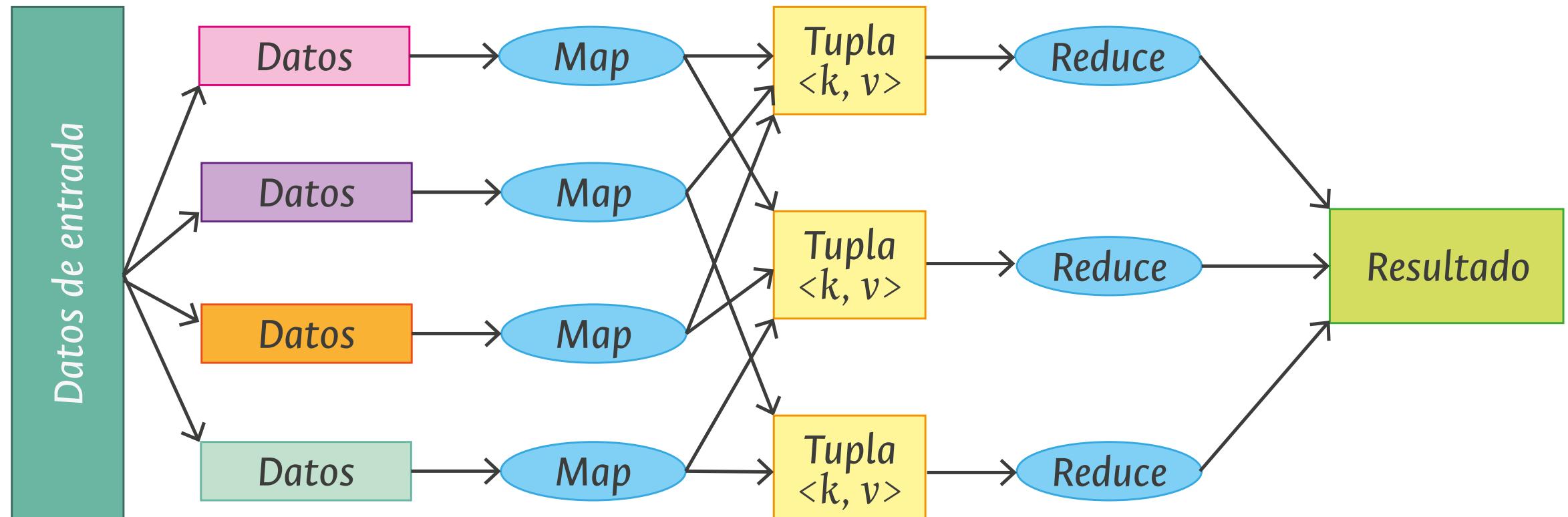
Las **sharding keys** son las claves indexadas que usa MongoDB para partir las colecciones fragmentadas y distribuirlas en el clúster fragmentado. Estas claves sirven para crear agrupaciones lógicas de documentos que se distribuyen entre los distintos shards. A cada uno de estos grupos lógicos se le denomina **chunk**.

**MongoDB** **Sharding (Video explicativo)**

10

## Map – Reduce

- Map – Reduce es un modelo de datos creado para brindar soporte en la computación paralela y el manejo de grandes conjuntos de datos en ciertos problemas específicos.
- Propuesto y empleado por Google para mejorar rendimiento en la clasificación de las páginas web que se muestran en el navegador.



**Map - Reduce****Funciones**

Se usa en Big Data por su capacidad para dividir y procesar en paralelo grandes cantidades de información. Debido a los grandes volúmenes que se manejan con este modelo, se acostumbra a ejecutarlo sobre sistemas de archivos distribuidos.

**Función Map**

Esta función toma los datos de entrada y los divide en bloques más pequeños en forma de pares llave – valor. Luego, procesa cada una y genera cero o más salidas de llave – valor. Esto se aplica a cada elemento de la entrada, por lo que se obtendrá una lista de pares por cada llamada a la función

**Función Reduce**

Se ejecuta en paralelo y, por cada llave única, la función itera a través de la lista de todos los valores que corresponden a la misma. No suele devolver una colección de documentos; más bien, devuelve un nuevo documento que contiene la información derivada.

# MongoDB



**MongoDB**



**Operaciones de agregación en MongoDB: con la función *aggregate* y con la función *mapReduce***



## MongoDB

## Actividades



# mongoDB



Taller guiado de MongoDB y Python.



Taller guiado de Conceptos de MongoDB.



Taller guiado de Map-Reduce y Operaciones de agregación



**Taller III - MongoDB.**



## Despedida

**¡Gracias por su atención!**

**Jorge Eliecer Camargo  
Mendoza, PhD.**

<https://dis.unal.edu.co/~jecamargom/>

[jecamargom@unal.edu.co](mailto:jecamargom@unal.edu.co)



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

Departamento de Ingeniería de Sistemas e Industrial

Facultad de Ingeniería

Universidad Nacional de Colombia

Sede Bogotá



## Referencias

- Chodorow, K., Dirolf, M. (2010). MongoDB: The Definitive Guide. (1era edición). Recuperado de <https://n9.cl/g64d1>
- Meier, A., Kaufmann, M. (2019). The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing. (1era edición). Recuperado de <https://n9.cl/9qg76>
- Gandini, A., Gribaudo, M., Knottenbelt, W., Osman, R. and Piazzolla, P. (2018). NoSQL Database Architecture: MongoDB. *Performance evaluation of NoSQL databases*. Politecnico di Milano, 34 (5). Recuperado de <https://www.doc.ic.ac.uk/~wjk/publications/gandini-gribaudo-knottenbelt-osman-piazzola-epew-2014.pdf>
- Bases de datos documentales: ¿cómo funciona un document store?. (23 de marzo de 2020). IONOS. <https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/bases-de-datos-documentales/>
- Cloud Firestore. (11 de agosto de 2022). Goofle Firebase. <https://firebase.google.com/docs/firestore>
- ¿Qué es Azure Cosmos DB?. (9 de febrero de 2021). Govms. <https://www.govms.cl/que-es-cosmos-db/>
- JSON and BSON. (7 de septiembre de 2022). MongoDB. <https://www.mongodb.com/json-and-bson>
- ¿Qué es el sharding en MongoDB? ¿Cómo funciona el sharding en MongoDB?. (4 de julio de 2019). Ramon Carrasco. <https://www.ramoncarrasco.es/es/content/es/kb/141/que-es-el-sharding-en-mongodb-como-funciona-el-sharding-en-mongodb>
- ¿Qué es el sharding en MongoDB? ¿Cómo funciona el sharding en MongoDB?. (4 de julio de 2019). Ramon Carrasco. <https://www.ramoncarrasco.es/es/content/es/kb/143/que-son-las-shard-keys-en-mongodb-como-se-crea-una-sharded-database-en-mongodb>



## Recursos adicionales

- MongoDB. inc. (s. f.). MongoDB University. <https://university.mongodb.com/>



## Derechos de imágenes

- Alexander Sosluev (s.f) Mongodb.png [Logo]. <https://commons.wikimedia.org/wiki/File:Mongodb.png>
- Database. (s.f.). existdb.png [Logo]. <https://dbdb.io/media/logos/existdb.png>
- TopPNG. (s.f.). mongodb.png [Logo]. <https://n9.cl/7fte9>
- Christian Grün. (s.f.). BaseX\_512px.png [Logo]. [https://upload.wikimedia.org/wikipedia/commons/6/6e/BaseX\\_512px.png](https://upload.wikimedia.org/wikipedia/commons/6/6e/BaseX_512px.png)
- PNGitem. (s.f.). firebase-firebase-png-transparent.png [Logo]. <https://n9.cl/fua31>
- Bob Pusateri. (s.f.). CosmosDB-400.png [Logo]. <https://n9.cl/8f05e>
- Amazon Web Services LLC. (2 de diciembre de 2011). AWS\_Simple\_Icons\_Database\_Amazon\_SimpleDB\_Item.svg.png [Logo].  
[https://en.wikipedia.org/wiki/File:AWS\\_Simple\\_Icons\\_Database\\_Amazon\\_SimpleDB\\_Item.svg](https://en.wikipedia.org/wiki/File:AWS_Simple_Icons_Database_Amazon_SimpleDB_Item.svg)
- [https://commons.wikimedia.org/wiki/File:Apache\\_CouchDB\\_logo.svg](https://commons.wikimedia.org/wiki/File:Apache_CouchDB_logo.svg)
- Apache Software Foundation. (12 de julio de 2018). Apache\_CouchDB\_logo.svg.  
[https://commons.wikimedia.org/wiki/File:Apache\\_CouchDB\\_logo.svg](https://commons.wikimedia.org/wiki/File:Apache_CouchDB_logo.svg)



## Derechos de imágenes

- 1000 Marcas (1 de junio 2022) Cisco.png [Logo]. <https://1000marcas.net/logo-cisco/>
- Bosch. (s.f.) Bosch\_logo.png [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:Bosch\\_logo.png](https://es.m.wikipedia.org/wiki/Archivo:Bosch_logo.png)
- Forbes Inc. (1 de febrero de 2014). Forbes\_logo.svg [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:Forbes\\_logo.svg](https://es.m.wikipedia.org/wiki/Archivo:Forbes_logo.svg)
- 1000 Marcas. (17 de octubre 2022) CERN-Logo-500x423.png [Logo]. <https://1000marcas.net/cern-logo/>
- Lippincott Studio, Adrian Frutiger. (Octubre de 2012). 300px-EBay\_logo.svg.png [Logo].  
[https://es.m.wikipedia.org/wiki/Archivo:EBay\\_logo.svg](https://es.m.wikipedia.org/wiki/Archivo:EBay_logo.svg)
- Amazon.com, Inc. (6 de agosto de 2016). Amazon\_logo.svg [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:Amazon\\_logo.svg](https://es.m.wikipedia.org/wiki/Archivo:Amazon_logo.svg)
- 1000 Marcas. (17 de octubre 2022) Logo-Linkedin-500x313.png [Logo]. <https://1000marcas.net/linkedin-logo/>
- MetLife. (s.f.). MetLife.png [Logo]. <https://jobs.lever.co/metlife>
- Microsoft Corporation. (s.f.). Microsoft\_Azure\_Logo.svg [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:Microsoft\\_Azure\\_Logo.svg](https://es.m.wikipedia.org/wiki/Archivo:Microsoft_Azure_Logo.svg)
- Database. (s.f.). existdb.png [Logo]. <https://dbdb.io/media/logos/existdb.png>
- TopPNG. (s.f.). mongodb.png [Logo]. <https://n9.cl/7fte9>
- McAfee. (s.f.). McAfee Logo 2017.png [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:McAfee\\_Logo\\_2017.png](https://es.m.wikipedia.org/wiki/Archivo:McAfee_Logo_2017.png)



## Derechos de imágenes

- Wal-Mart Stores, Inc. (3 de julio de 2008). Walmart logo.svg. [https://commons.wikimedia.org/wiki/File:Walmart\\_logo.svg](https://commons.wikimedia.org/wiki/File:Walmart_logo.svg)
- Ed Benguiat. (21 de febrero de 1967). The New York Times Logo.svg [Logo].  
[https://es.wikipedia.org/wiki/Archivo:The\\_New\\_York\\_Times\\_Logo.svg](https://es.wikipedia.org/wiki/Archivo:The_New_York_Times_Logo.svg)
- Air France. (s.f.). Air France Logo.svg [Logo]. [https://es.m.wikipedia.org/wiki/Archivo:Air\\_France\\_Logo.svg](https://es.m.wikipedia.org/wiki/Archivo:Air_France_Logo.svg)
- Flaticon. (s.f.). Complain free icon. [Icono]. [https://www.flaticon.com/free-icon/complain\\_1041891](https://www.flaticon.com/free-icon/complain_1041891)
- Flaticon. (s.f.). Json File free icon. [Icono]. [https://www.flaticon.com/free-icon/json-file\\_136525](https://www.flaticon.com/free-icon/json-file_136525)
- Flaticon. (s.f.). File free icon. [Icono]. [https://www.flaticon.com/free-icon/file\\_473685](https://www.flaticon.com/free-icon/file_473685)
- Flaticon. (s.f.). Plus free icon. [Icono]. [https://www.flaticon.com/free-icon/plus\\_753317](https://www.flaticon.com/free-icon/plus_753317)
- Flaticon. (s.f.). Minus free icon. [Icono]. [https://www.flaticon.com/free-icon/minus\\_753340](https://www.flaticon.com/free-icon/minus_753340)
- Flaticon. (s.f.). Search free icon. [Icono]. [https://www.flaticon.com/free-icon/search\\_607554](https://www.flaticon.com/free-icon/search_607554)
- Flaticon. (s.f.). Upgrade free icon. [Icono]. [https://www.flaticon.com/free-icon/upgrade\\_2512773](https://www.flaticon.com/free-icon/upgrade_2512773)
- Flaticon. (s.f.). Data server free icon [Icono]. [https://www.flaticon.com/free-icon/data-server\\_5960357](https://www.flaticon.com/free-icon/data-server_5960357)



## Créditos

Facultad de  
**INGENIERÍA**

**Autores**

Jorge Eliécer Camargo Mendoza, PhD

**Asistente docente**

Juan Sebastián Lara Ramírez

Edder Hernández Forero

Brian Chaparro Cetina

Rosa Alejandra Superlano Esquivel

Leonardo Avendaño Rocha

Alberto Nicolai Romero Martínez

**Diseño instruccional**

Claudia Patricia Rodríguez Sánchez

**Diseño gráfico**

Clara Valeria Suárez Caballero

Milton R. Pachón Pinzón

**Diagramadora PPT**

Daniela Duque

**Diseño de imágenes**

Rosa Alejandra Superlano Esquivel

**Fecha**

2022-II

