



mongoDB

# CONOCIENDO MONGO DB.

M. EN C. ÁNGEL SALVADOR LÓPEZ VÁSQUEZ  
ARTÍCULO MDB. AÑO 2014.

<b>Introducción</b> .....	2
<b>¿Cómo funciona Mongo DB?</b> .....	4
<b>¿Dónde se puede utilizar MongoDB?</b> .....	6
<b>¿Dónde no se debe usar MongoDB?</b> .....	7
<b>¿Cómo se instala MongoDB?</b> .....	8
<b>¿Cómo puedo usar la consola?</b> .....	8

## Introducción



Dentro de las bases de datos NoSQL, probablemente una de las más famosas sea **Mongo DB**. Con un concepto muy diferente al de las bases de datos relacionales, se está convirtiendo en una interesante alternativa.

Pero cuando uno se inicia en MongoDB se puede sentir perdido. No tenemos tablas, no tenemos registros y lo que es más importante, no tenemos SQL. Aun así, MongoDB es una seria candidata para almacenar los datos de nuestras aplicaciones.

## ¿Qué es MongoDB?

Es una base de datos orientada a documentos. Esto quiere decir que en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en BSON, que es una representación binaria de JSON.

Una de las diferencias más importantes con respecto a las bases de datos relacionales, es que **no es necesario seguir un esquema**. Los documentos de una misma colección concepto similar a una tabla de una base de datos relacional, pueden tener esquemas diferentes.

Imaginemos que tenemos una colección a la que llamamos Personas. Un documento podría almacenarse de la siguiente manera:

```
{
  Nombre: "Pedro",
  Apellidos: "Martínez Campo",
  Edad: 22,
  Aficiones: ["fútbol", "tenis", "ciclismo"],
  Amigos: [
    {
      Nombre: "María",
      Edad: 22
    },
    {
      Nombre: "Luis",
      Edad: 28
    }
  ]
}
```

El documento anterior es un clásico documento JSON. Tiene strings, arrays, subdocumentos y números. En la misma colección podríamos guardar un documento como este:

```
{
  Nombre: "Luis",
  Estudios: "Administración y Dirección de
Empresas",
  Amigos:12
}
```

Este documento no sigue el mismo esquema que el primero. Tiene menos campos, algún campo nuevo que no existe en el documento anterior e incluso un campo de distinto tipo.

Esto que es algo impensable en una base de datos relacional, es algo totalmente válido en **MongoDB**.

## ¿Cómo funciona Mongo DB?

MongoDB está escrito en C++, aunque las consultas se hacen pasando objetos JSON como parámetro. Es algo bastante lógico, dado que los propios documentos se almacenan en BSON. Por ejemplo:

```
db.Clientes.find({Nombre:"Pedro"});
```

La consulta anterior buscará todos los clientes cuyo nombre sea Pedro.

Mongo DB viene de serie con una consola desde la que podemos ejecutar los distintos comandos. Esta consola está construida sobre JavaScript, por lo que las consultas se realizan utilizando ese lenguaje. Además de las funciones de Mongo DB, podemos utilizar muchas de las funciones propias de JavaScript. En la consola también podemos definir variables, funciones o utilizar bucles.

Si queremos usar nuestro lenguaje de programación favorito, existen drivers para un gran número de ellos. Hay drivers oficiales para C#, Java, Node.js, PHP, Python, Ruby, C, C++, Perl o Scala. Aunque estos drivers están soportados por Mongo DB, no todos están en el mismo estado de madurez. Por ejemplo el de C es una versión alpha. Si queremos utilizar un lenguaje concreto, es mejor revisar los drivers disponibles para comprobar si son adecuados para un entorno de producción.

## ¿Dónde se puede utilizar MongoDB?

Aunque se suele decir que las bases de datos NoSQL tienen un ámbito de aplicación reducido, **MongoDB se puede utilizar en muchos de los proyectos que desarrollamos en la actualidad.**

Cualquier aplicación que necesite almacenar datos semi estructurados puede usar **MongoDB**. Es el caso de las típicas aplicaciones CRUD o de muchos de los desarrollos web actuales.

Eso sí, aunque las colecciones de **MongoDB** no necesitan definir un esquema, es importante que diseñemos nuestra aplicación para seguir uno. Tendremos que pensar si necesitamos normalizar los datos, de normalizarlos o utilizar una aproximación híbrida. Estas decisiones pueden afectar al rendimiento de nuestra aplicación. En definitiva el esquema lo definen las consultas que vayamos a realizar con más frecuencia.

MongoDB es especialmente útil en entornos que requieran escalabilidad. Con sus opciones de replicación y sharding, que son muy sencillas de configurar, podemos conseguir un sistema que escale horizontalmente sin demasiados problemas.

## ¿Dónde no se debe usar MongoDB?

En esta base de datos **no existen las transacciones**. Aunque nuestra aplicación puede utilizar alguna técnica para simular las transacciones, **MongoDB** no tiene esta capacidad. Solo garantiza operaciones atómicas a nivel de documento. Si las transacciones son algo indispensable en nuestro desarrollo, deberemos pensar en otro sistema.

**Tampoco existen los JOINS**. Para consultar datos relacionados en dos o más colecciones, tenemos que hacer más de una consulta. En general, si nuestros datos pueden ser estructurados en tablas, y necesitamos las relaciones, es mejor que optemos por un RDBMS clásico.

Y para finalizar, están las consultas de agregación. **MongoDB** tiene un framework para realizar consultas de este tipo llamado Aggregation Framework. También puede usar Map Reduce. Aún así, estos métodos no llegan a la potencia de un sistema relacional. Si vamos a necesitar explotar informes complejos, deberemos pensar en utilizar otro sistema. Eso sí, esta es una brecha que MongoDB va recortando con cada versión. En poco tiempo esto podría dejar de ser un problema.



## ¿Cómo se instala MongoDB?

La instalación de una instancia del servidor es un juego de niños. Simplemente tenemos que bajar los binarios para nuestro sistema operativo. Hay versiones par Windows, Linux y MacOS. Una vez bajados podremos arrancar el servicio de MongoDB con un solo comando.

```
mongod --dbpath data
```

Con este comando arrancamos el servicio mongod, que empezará a escuchar peticiones por el puerto 27017.

Es importante indicar el parámetro “*dbpath*”, con la ruta dónde se almacenarán los ficheros de nuestra base de datos.

## ¿Cómo puedo usar la consola?

Si ya tenemos el servidor lanzado en nuestra máquina, bastará con lanzar desde la consola el siguiente comando.

```
mongo localhost
```

Desde ese momento entraremos en la consola y podremos realizar consultas. Si escribimos “*help*” tendremos un listado con los comandos más comunes y su descripción.