

UNIVERSIDAD PRIVADA DE TACNA



INGENIERIA DE SISTEMAS

TITULO:

Trabajo-Encargado-N-05-SQL-y-NoSQL

CURSO:

BASE DE DATOS II

DOCENTE(ING):

Patrick Cuadros Quiroga

Integrantes:

Adnner Esperilla Ruiz
Wilfredo Vilca Chambilla

(2015050543)
(2006028540)

2019-Tacna

Comparación BD NoSQL

Resumen

Se hará la comparación de una base de datos NoSQL, dando a conocer sus características principales y analizando lo importante que son sus definiciones de los tipos de base de datos NoSQL con el fin de brindar conocimiento con que base de datos pueda trabajar según sus respectivas áreas, también veremos la creación de base de datos, inserción y consultas de datos NoSQL mediante la plataforma Docker.

Palabras clave: NoSQL, MySQL, Docker.

Abstract

The comparison of a NoSQL database will be done, making known its main characteristics, and analyzing what your NoSQL database will be in order to provide knowledge about the database in order to work with your reactive areas, we will also see the creation of the database, insertion and queries of NoSQL data through the Docker platform.

I. INTRODUCCIÓN

Ahora en nuestros tiempos que viene ser la era digital el manejo de la información se hace más complejo ya que la cantidad de datos es abrumadora; ya que por motivos mayores las personas buscan tecnologías que le ayuden a simplificar estos problemas. Las bases de datos relacionales son las más comunes, pero en los últimos años ha aumentado el interés por las bases de datos NoSQL, que son un conjunto de tecnologías que ayudan el manejo de información y su distribución según la necesidad de las personas.

Entonces el presente documento hace un análisis y revisión de las tecnologías NoSQL, haciendo posible hacer una comparación fácil de comprender ante cualquier usuario con conocimiento mínimo en lo que es base de datos.

El resto de este artículo está organizado de la siguiente manera. En la Sección 2 se muestran los requisitos necesarios y métodos usados para el desarrollo de este artículo. La Sección 3 se explican los resultados. Y finalmente, las conclusiones están en la Sección 4.

II. REQUISITOS Y MÉTODOS

A. Materiales

- Tener activado la virtualización en el BIOS
- Para poder ejecutar contenedores Docker es necesario tener instalado Docker Community Edition (CE) en nuestro equipo
- Windows 10 64bit: Pro, Enterprise o Education, con al menos 4GB de RAM.

B. Métodos

- Se utilizó como material artículos y libros relacionados a la base de datos NoSQL, repositorios, páginas web.

III. MARCO TEÓRICO

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos. Coniene las siguientes características:

- Se emplean métodos determinados para incluir datos nuevos y para borrar, modificar o recuperar los datos almacenados.
- Los datos son independientes de los programas que los usan.
- Los datos están interrelacionados, sin redundancias innecesarias.

A. Tipos de base de datos

- Hay bases de datos relacionales, como MySQL, SQL Server y Oracle. Como su nombre lo indica utilizan el modelo relacional y siempre es mejor usarlas cuando los datos son consistentes y ya tienes algo planificado.
- También existen las no relacionales, como MongoDB y Redis, conocidas como NO-SQL (Not Only SQL). Estas son más flexibles en cuanto a consistencia de datos y se han convertido en una opción que intenta solucionar algunas limitaciones que tiene el modelo relacional.
- Además hay otras BBDD no tan tradicionales, como las basadas en grafos o aquellas que tienen información cartográfica, que pueden servir, por ejemplo, si estás creando un e-commerce para encontrar relaciones entre los productos y las preferencias de los usuarios.

B. Bases de datos relacional

- La interfaz estándar de programa de usuario y aplicación a una base de datos relacional es el lenguaje de consultas estructuradas (SQL)
- Una base de datos relacional es un conjunto de tablas que contienen datos provistos en categorías predefinidas.
- Ejemplo modelo relacional:



C. Modelo NoSQL

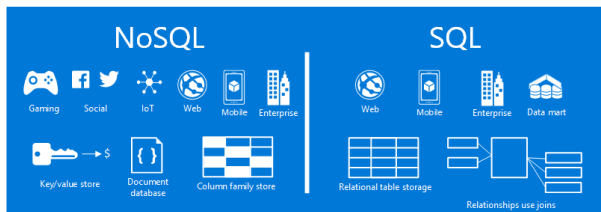
Las bases de datos NoSQL utilizan una variedad de modelos de datos para acceder y administrar datos, como documentos, gráficos, clave-valor, en-memoria y búsqueda. Estos tipos de bases de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas de las restricciones de coherencia de datos en otras bases de datos.



En una base de datos NoSQL, el registro de un libro generalmente se almacena como un documento JSON. Para cada libro, el elemento, ISBN, Título del libro, Número de edición, Nombre autor y IDAutor se almacenan como atributos en un solo documento. En este modelo, los datos están optimizados para un desarrollo intuitivo y escalabilidad horizontal.

as bases de datos NoSQL se adaptan perfectamente a muchas aplicaciones modernas, como dispositivos móviles, web y juegos, que requieren bases de datos flexibles, escalables, de alto rendimiento y altamente funcionales para proporcionar excelentes experiencias de usuario.

- **Flexibilidad:** Las bases de datos NoSQL generalmente ofrecen esquemas flexibles que permiten un desarrollo más rápido y más iterativo.
- **Escalabilidad:** las bases de datos NoSQL generalmente están diseñadas para escalar usando clústeres distribuidos de hardware en lugar de escalar añadiendo servidores caros y sólidos.
- **Alto rendimiento:** La base de datos NoSQL está optimizada para modelos de datos específicos y patrones de acceso que permiten un mayor rendimiento que el intento de lograr una funcionalidad similar con bases de datos relacionales.
- **Altamente funcional:** Las bases de datos NoSQL proporcionan API altamente funcionales y tipos de datos que están diseñados específicamente para cada uno de sus respectivos modelos de datos.



IV. RESULTADOS

A. Creacion de base de datos NoSQL con MySql

- **MySql** es una base NoSQL orientada a objetos
- **Arquitectura cliente y servidor:** MySQL, al igual que cualquier otro sistema de registros de datos, es un programa de registro basado en un sistema entre cliente y servidor.
- **Compatibilidad con SQL:** SQL es un lenguaje de programación que permite tanto la consulta como la renovación de datos para la gestión de una base en la que se almacena un conjunto de datos.
- **Lenguaje de programación:** la base de datos MySQL está escrita en C y C++, dos de los lenguajes de programación más demandados y populares de todo el mundo.
- **Sistemas de almacenamiento:** este tipo de bases proporciona sistemas de almacenamiento tanto transaccionales como no transaccionales.

B. Instalar MySql en Docker

- Ingresmos nuestra cuenta creada en Docker Hub para iniciar sesión en el aplicativo. Ubicar la aplicación PowerShell, ejecutarla como Administrador.

En la ventana de comandos de PowerShell escribir lo siguiente:



- Lo primero que hay que hacer es descargar el contenedor de MySQL con el siguiente comando.

```
PS C:\Users\Esperrilla Ruiz> $ docker run -d --p 33060:3306 --name mysql-db --e MYSQL_ROOT_PASSWORD=secret mysql
```

- Con esta tenemos nuestro contenedor escuchando

| CONTAINER ID | IMAGE | CREATED | STATUS | PORTS | NAMES |
|--------------|-------|---------------|--------------|------------------------------------|----------|
| b62cae4104ed | mysql | 5 minutes ago | Up 5 minutes | 33060/tcp, 0.0.0.0:33060->3306/tcp | mysql-db |

- Para entrar al contenedor usamos un modo interactivo para asignar un TTY (terminal) y un STDIN abierto

```
PS C:\Users\Esperrilla Ruiz> $ docker exec -it mysql-db mysql -p
```

```
$ docker exec -it mysql-db mysql -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.13 MySQL Community Server - GPL

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database demo;
Query OK, 1 row affected (0.26 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| demo     |
| information_schema |
| mysql    |
| performance_schema |
| sys      |
+-----+
5 rows in set (0.00 sec)

mysql>
```

- Eliminamos el proceso que corre el contenedor creado.

```
$ docker rm -f mysql-db
```

- Eliminamos todos los volúmenes ya que Docker crea volúmenes temporales sin pedirte permiso.

```
$ docker volume prune
```

- Creamos un volumen

```
$ docker volume create mysql-db-data
```

- Verificamos que se haya creado el volumen

```
$ docker volume ls
DRIVER      VOLUME NAME
local       mysql-db-data
```

- Levantamos nuevamente el Docker y agregamos el volumen con la opción `--mount`

```
PS C:\Users\Esperrilla Ruiz> $ docker run -d -p 33060:3306 --name mysql-db --e MYSQL_ROOT_PASSWORD=secret --mount src=mysql-db-data dst=/var/lib/mysql mysql
```

- Entramos al contenedor de forma interactiva o desde el Workbench y creamos una base de datos

```
$ docker exec -it mysql-db mysql -p
...
mysql> create database demo;
Query OK, 1 row affected (0.32 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| demo     |
| information_schema |
| mysql    |
| performance_schema |
| sys      |
+-----+
5 rows in set (0.00 sec)

mysql>
```

- Terminamos el proceso

```
PS C:\Users\Esperrilla Ruiz> $ docker rm -f mysql-db
```

C. Inserción y consulta de datos

- Inserción de datos en la tabla en MySQL

```
mysql> mysql> insert into usuarios(id,nombre,apellido) VALUES (1,juan,Miane);
qweqwe Query OK, 1 row affected (0.07 sec)
```

- Se hace copia el siguiente código para poder usar la consulta en MySQL de una tabla

```
mysql> select*from usuarios;
+-----+-----+-----+
| id | nombre | apellido |
+-----+-----+-----+
| 1 | Juan | Miane |
| 2 | María | Ramirez |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

D. Comparación entre MySQL y Oracle

MySQL

- MySQL es una fuente abierta y MySQL está disponible para su descarga e instalación gratuitas.
- La autenticación del usuario se realiza en MySQL usando solo la ubicación, el nombre de usuario y la contraseña.
- La flexibilidad de crear procedimientos y funciones almacenados utilizando PL / SQL es muy inferior en MySQL.
- MySQL ofrece muy pocos comandos relacionados con la generación de resultados como informe y la definición de variables. MySQL incluye solo comandos SQL muy simples.
- MySQL no tiene la función de bóveda de auditoría en el servidor..
- MySQL no ofrece herramientas a nivel empresarial.
- MySQL solo tiene facilidad de bloqueo de tablas.
- MySQL no tiene amplias funciones de almacenamiento como tablespace, sinónimo, paquetes y muchos otros.
- La base de datos MySQL no soporta XML.
- MySQL solo admite dos tipos de caracteres, a saber, CHAR y VARCHAR.
- MySQL, las tablas temporales solo son visibles dentro de la sesión activa actual. Cuando la sesión expira, las tablas temporales se eliminan automáticamente.
- MySQL solo tiene dos mecanismos de copia de seguridad: `mysqlhotcopy` y `mysqldump`.

Oracle

- Solo Oracle Express Edition es gratuito. Pero Oracle Express Edition tiene características muy limitadas en comparación con MySQL

- Oracle proporciona seguridad de base de datos mejorada. La autenticación de usuarios se realiza en Oracle especificando roles globales además de la ubicación, el nombre de usuario y la contraseña
- Oracle proporciona funciones más flexibles para crear procedimientos y funciones almacenados utilizando PL / SQL.
- Oracle incluye comandos SQL extensos en SQL * Plus, incluidos comandos para generar resultados como informe y definir variables.
- Oracle incluye comandos SQL extensos en SQL * Plus, incluidos comandos para generar resultados como informe y definir variables.
- Oracle proporciona instalaciones de bóveda de auditoría.
- Oracle ofrece herramientas a nivel empresarial.
- Oracle proporciona la facilidad de bloqueo de fila también.
- Oracle tiene unas características de almacenamiento muy extensas. Oracle admite tablespace, sinónimo, paquetes y todas las demás características.
- Oracle soporta y usa XML.
- Oracle admite cuatro tipos de datos de caracteres diferentes, a saber: CHAR, VARCHAR2, NCHAR, NVARCHAR2.
- Oracle ofrece muchos mecanismos de copia de seguridad, incluida la copia de seguridad en caliente, copia de seguridad, importación, exportación y muchos otros.

E. Documental

Una base de datos documental, también llamada una base de datos orientada a documentos u tienda de documentos, es un subconjunto de un tipo de base de datos NoSQL.

Algunos almacenes de documentos también pueden ser bases de datos de valores clave. Una base de datos de documentos se utiliza para almacenar, recuperar y administrar datos semiestructurados.

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.

El esquema puede variar, proporcionando mucha más flexibilidad para el modelado de datos que las bases de

datos relacionales.

Las bases de datos documental almacena cada registro y sus datos asociados en un solo documento. Cada documento contiene datos semiestructurados que pueden ser consultados con el uso de varias herramientas de consulta y análisis del DBMS.

Ejemplo en XML:

```
<artist>
  <artistname>Iron Maiden</artistname>
  <albums>
    <album>
      <albumname>The Book of Souls</albumname>
      <datereleased>2015</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Killers</albumname>
      <datereleased>1981</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Powerslave</albumname>
      <datereleased>1984</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Somewhere in Time</albumname>
      <datereleased>1986</datereleased>
      <genre>Hard Rock</genre>
    </album>
  </albums>
</artist>
```

Fundamentos del formato JSON :

- Pares de valores clave o atributos : Con el formato JSON se almacena en un par de valores clave. A estos pares se les llama a veces atributos. Las claves son cadenas simples y los valores pueden ser de cualquier tipo.
- Incrustación de objetos: Los valores incluidos en el par de valores clave también pueden ser otros objetos JSON, lo que permite crear una jerarquía de objetos. Colocar objetos JSON dentro de otro objeto JSON se denomina modelo de datos incrustados en base de datos documentales.
- Matrices: El formato JSON trabaja con un lenguaje de programación natural en todos los lenguajes de programación y estructuras de datos que son las matrices, por lo cual se admite almacenamiento de matrices como valores contra una clave.

Ejemplo en JSON:

```
{
  '_id' : 1,
  'artistName' : { 'Iron Maiden' },
  'albums' : [
    {
      'albumname' : 'The Book of Souls',
      'datereleased' : 2015,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Killers',
      'datereleased' : 1981,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Powerslave',
      'datereleased' : 1984,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Somewhere in Time',
      'datereleased' : 1986,
      'genre' : 'Hard Rock'
    }
  ]
}
```

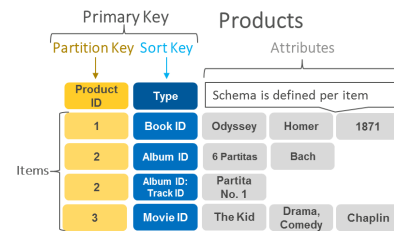
En la imagen podemos ver un ejemplo de un documento de tipo JSON describe un libro.[?]

Beneficios :

- Modelado flexible de datos: a medida que las aplicaciones web, móviles, sociales e IoT cambian la naturaleza de los modelos de datos de aplicaciones, las bases de datos de documentos eliminan la necesidad de forzar modelos de datos relacionales para admitir nuevos tipos de modelos de datos de aplicaciones.
- Rendimiento de escritura rápido: a diferencia de las bases de datos relacionales tradicionales, algunas bases de datos de documentos priorizan la disponibilidad de escritura sobre la estricta consistencia de los datos. Esto garantiza que las escrituras siempre serán rápidas, incluso si una falla en una parte del hardware o de la red da como resultado un pequeño retraso en la replicación de datos y la coherencia en todo el entorno.
- Rendimiento rápido de consultas: muchas bases de datos de documentos tienen potentes motores de búsqueda y funciones de indexación que proporcionan capacidades de consulta rápidas y eficientes.

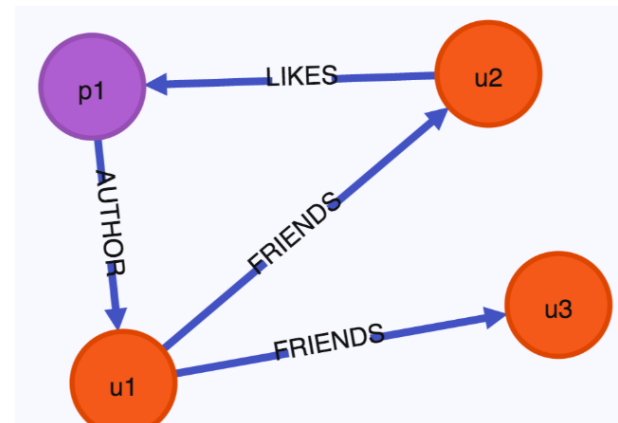
F. Clave-Valor

- Una base de datos clave-valor es un tipo de base de datos no relacional que utiliza un método simple de clave-valor para almacenar datos. Una base de datos clave-valor almacena datos como un conjunto de pares clave-valor en los que una clave sirve como un identificador único.



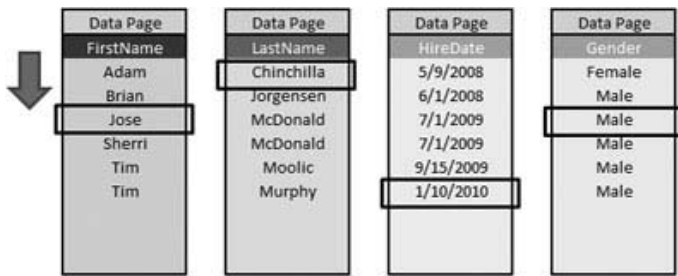
G. Grafos

- En nuestras base de datos se puede utilizar el modelo de Grafos para ayudar .
- Disponer de más información con agilidad y eficiencia
- Las podemos utilizar por ejemplo para guardar puntos de un camino, relaciones de amigos, familia, o cualquier tipo de dato que represente alguna relación
- Consultas más amplias y no demarcadas por tablas



H. Tabular (Column-Store)

- Las ventajas de rendimiento en los índices de almacén de columnas son posibles al aprovechar la tecnología de compresión VertiPaq, que permite que grandes cantidades de datos se compriman en la memoria.
- Lo datos son almacenados en columnas.
- En una columna tiene múltiples datos.
- Un índice de almacén de columnas es más eficiente para este ejemplo porque solo se necesita una página de datos (comprimidos) de menor tamaño para satisfacer la consulta.



| Data Page | Data Page | Data Page | Data Page |
|-----------|------------|-----------|-----------|
| FirstName | LastName | HireDate | Gender |
| Adam | Chinchilla | 5/9/2008 | Female |
| Brian | Jorgensen | 6/1/2008 | Male |
| Jose | McDonald | 7/1/2009 | Male |
| Sherri | McDonald | 7/1/2009 | Male |
| Tim | Moolic | 9/15/2009 | Male |
| Tim | Murphy | 1/10/2010 | Male |

V. DISCUSIÓN Y CONCLUSIONES

- NoSQL permite el manejo de grandes volúmenes de datos y la posibilidad de tener un sistema distribuido.
- La selección de la tecnología de almacenamiento adecuada involucra la consideración de numerosos aspectos.
- Una de las aplicaciones de la teoría de grafos se aplica en el almacenamiento de grandes cantidades de información, como por ejemplo la almacenada en una red social.
- Cada vez con más frecuencia estamos viendo cómo las tecnologías NoSQL forman parte de la solución en proyectos empresariales, gracias a beneficios como la mejora en la productividad de los equipos de desarrollo, y la posibilidad de llegar antes al mercado y con una considerable reducción del TCO.

Bibliografía

<https://www.uaeh.edu.mx/docencia/Tesis/icbi/maestria/documento>
<http://repositorio.espm.edu.ec/bitstream/42000/74/1/TESIS>
<http://repositorio.ug.edu.ec/bitstream/redug/24100/1/B-CISC-PTG.1382.Balladares>
<https://www.paradigmadigital.com/techbiz/breve-introduccion-las-tecnologias-nosql/>
<https://smarterworkspaces.kyocera.es/blog/las-bases-datos-documentales/>
<https://www.javeriana.edu.co/biblos/tesis/ciencias/tesis572.pdf>