**Universidad Centroamericana “José Simeón Cañas“**

****

***NoSQL: Base de datos clave-valor sobre el gestor Riak***

**Integrantes:**

* Luis Eduardo Cortez Murillo 00029117
* Daniel Leonardo Hernández Chica 00209317
* Fabricio Enrique Hernández López 00163217
* Luisa Daniela Arevalo Rodas 00038617
* Jose David Mejia Segura 00001617

**Catedrático:** Lic. Erick Giovanni Varela Guzman.

I. ÍNDICE

Introducción ------------------------------------------------------------------------------------- pág. 2

Marco Teórico ----------------------------------------------------------------------------------- pág. 3

Conclusiones ------------------------------------------------------------------------------------ pág. 6

Referencias Bibliográficas -------------------------------------------------------------------- pág. 7

Anexos ------------------------------------------------------------------------------------------- pág. 8

II. INTRODUCCIÓN

Una base de datos de valores-clave (también conocida como almacén de valores-clave y base de datos key-value) es un tipo de base de datos NoSQL que utiliza un método simple de clave / valor para almacenar datos.

La parte clave-valor se refiere al hecho de que la base de datos almacena datos como una colección de pares clave / valor. Este es un método simple de almacenar datos, y se sabe que escala bien.

El par clave-valor es un concepto bien establecido en muchos lenguajes de programación. Los lenguajes de programación normalmente se refieren a una clave-valor como una matriz asociativa o estructura de datos. Un valor-clave también se conoce comúnmente como diccionario o hash.

Un store de valores-clave o una base de datos de valores-clave es una base de datos simple que usa un arreglo asociativo (piensa en un mapa o diccionario) como el modelo de datos fundamental donde cada clave está asociada con un solo valor en una colección. Esta relación se conoce como un par clave-valor.

En cada par clave-valor, la clave se representa mediante una cadena arbitraria, como un nombre de archivo, URI o hash.

El valor puede ser cualquier tipo de datos como una imagen, archivo de preferencia del usuario o documento. El valor se almacena como un blob que no requiere modelado de datos inicial o definición de esquema.

III. MARCO TEÓRICO

***¿Qué es Riak?***

Riak es un almacén de datos clave-valor NoSQL distribuido que ofrece alta disponibilidad, tolerancia a fallas, simplicidad operativa y escalabilidad. Además de la versión de código abierto, viene en una versión empresarial compatible y una versión de almacenamiento en la nube [1]. Cuando hablamos de clave-valor se refiere a que cada elemento está identificado por una clave única, lo que permite la recuperación de la información de forma muy rápida, información que suele almacenarse como un objeto binario. Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras.[2]

Riak KV automatiza la distribución de datos en el clúster para lograr un rendimiento rápido y una sólida continuidad empresarial con una arquitectura sin maestro que garantiza una alta disponibilidad y se amplía de forma casi lineal utilizando el hardware básico para que pueda agregar capacidad fácilmente sin una gran carga operativa.

**Características de *Riak:***

Una de sus principales características es que es una base clave – valor escalable que simplifica el desarrollo, con él los usuarios se permiten formar prototipos, probar y desplegar aplicaciones de manera mucho más rápida que con otros modelos. Su diseño básicamente se enfoca en cuatro funciones: escalabilidad masiva, simplicidad, tolerancia a fallo y sencillez en las operaciones.

De entre estas las que más llaman la atención de los desarrolladores en Riak es que es tolerante a fallos, por lo que no pierden toda la información una vez haya un fallo, que consistente y tiene alta disponibilidad.

Riak se acoge a estos parámetros y como es un software libre muchos lo han empezado a utilizar dejando muy buenos comentarios respecto a su eficiencia.

La escalabilidad de la que presume es lineal y predecible, permite agregar nuevas máquinas sin mayor problema y sin que se cargue excesivamente el sistema.

La tolerancia a fallos es excepcional puesto que permite estar preparado en caso de daños a fin de no perder la información; su alta disponibilidad lo es tanto para lectura como para escritura siendo esta una característica garantizada; y la replicación, función con la cual Riakse puede configurar para ser replicado en varios cluster o diseñar cada uno con funciones diferentes.

***¿A qué se refiere una base de datos NoSQL?***

Una base datos NoSQL, también conocidas como “No sólo SQL”, nos referimos a una amplia clase de sistemas de gestión de datos (mecanismos para el almacenamiento y recuperación de datos) que difieren, en aspectos importantes, del modelo clásico de relaciones entre entidades (o tablas) existente en los sistemas de gestión bases de datos relacionales, siendo el más destacado el que no usan SQL como lenguaje principal de consulta [3].

Al igual que otros sistemas de gestión de información en metodología NoSQL, Riak llegó con la finalidad de ofrecer diferencias sistemáticas que lo pusieran por encima de sus compañeros entre las opciones de elección de los desarrolladores web, su gestión es muy buena al igual que la mayoría de la NoSQL pero se adapta a ciertas necesidades específicas según el tipo de web o de cliente al que va dirigido el tema.

Este entre otros ha utilizado a sus antecesores como muestra para mejorar los diversos aspectos que tras una u otra funcionalidad muestran aspectos a mejorar o pulir.

El sistema NoSQL se ha caracterizado por ofrecer valores agregados los modelos tradicionales en lenguaje SQL como un almacenamiento más alto, un mejor escalamiento y en general muchas otras características que ayudan de manera crucial el trabajo de desarrolladores y profesionales en el tema para llevar una verdadera gestión de información que permita cubrir las necesidades de los clientes para los que se esté implementando el modelo.

**Diferencias entre SQL y NoSQL**

**SQL.**

*Structured Query language* (SQL) o lenguaje de consulta estructurada, utiliza un estándar para manipular el contenido de las bases de datos relacionales. Cuenta con su propio lenguaje para consultar, insertar, actualizar y eliminar registros de la base de datos.

Se compone de una o varias tablas y cada tabla puede contener múltiples relaciones. En el siguiente elemento pongo como ejemplo en cómo está estructurada una sencilla base de datos por medio de tablas.

Las tablas representan la entidad de cada modelo de datos. Las relaciones permiten garantizar la persistencia de datos, evita la duplicidad de datos y controla la eliminación en cascada. Para ingresar un nuevo nuevo evento, bastaría con ingresar un nuevo registro en la tabla *events*.

SQL cuenta con un lenguaje de alto nivel que permite manipular la base de datos casi de cualquier forma. Por su manera de operar ayuda a mantener la integridad de los datos. Por ejemplo:

Para obtener la lista de todos los deportes

SELECT id, name FROM sports

Ingresar varios registros a la tabla categoría

INSERT INTO *categories* VALUES (*Varonil*, *Femenil*, *mixto*)

Su uso es más frecuente es en sistemas donde se necesite cuidar la integridad de datos como los bancos, sistemas de ventas, hospitales, CMS como WordPress.

Ejemplos de sistemas de gestión de base de datos son: MySQL, MariaDB, PostgreSQL.

## **Base de datos NoSQL**

También existen otros mecanismos de almacenamientos alternativos a las base de datos relacionales que se definen por tablas. Este tipo de base de datos, se apoyan en otros formatos. Los NOSQL son más adecuados para aquellos que manejan grandes volúmenes de datos. Es por eso que empresas como Facebook, Google, Amazon, Twitter, entro otros han popularizado estos tipos de base de datos.

Esas empresas tenían que enfrentarse a grandes desafíos motivados a la alta demanda de internet. Donde surge la necesidad de proporcionar información de grandes volúmenes de datos e indeterminado número de usuarios en el menor tiempo posible. Esas compañías se dieron cuenta que podían sacrificar la consistencia de los datos que garantiza una base de datos relacional para poder obtener un mejor rendimiento y ofrecer soluciones en tiempo real.

Los puedes usar en soluciones donde requieres interactuar con datos en tiempo real como en un chat, para análisis estadísticos donde procesas mucha información o incluso en blogs.

Algunos tipos de bases de datos NoSQL incluyen diferentes tipos de almacenamiento, por ejemplo: con columnas, documentos, key value store, gráficos, objetos (JSON), XML y otros modos de almacenamiento de datos.

## **¿Qué hace diferente SQL vs NOSQL?**

Una de las diferencias más relevantes entra una NOSQL a una base de datos relacional, es por su forma de almacenamiento no estructurada.

## **¿Cuál podría ser apropiada para mi proyecto?**

Depende mucho de la solución y cuales son los puntos fuertes que necesites ofrecer en tus gestión de información. Ambos tienen sus pros y sus contras. Puedes elegir entre la consistencia (SQL) o la alta disponibilidad (NOSQL). En la mayoría de mis proyectos apuesto por la consistencia porque los proyectos no requieren una disponibilidad en tiempo real de los datos.

¿Qué te pareció el artículo? Quedo al pendiente para tus comentario o cualquier duda. Si te pareció interesante, no dudes en compartirlo.

IV. CONCLUSIÓN

La conclusión a la que se llegó como grupo es que existen modelos de bases de datos que pueden aplicarse a una cantidad de problemas, por ejemplo, RIAK es una base de datos no relacional y es conveniente utilizarla cuando hablamos de “Big Data”. Otro ejemplo sería con el proyecto de bases de datos, sería más conveniente aplicar un modelo no relacional porque el tipo de almacenamiento conviene más para este tipo de problemas. Existen otro tipos de modelas para las bases de datos, como los orientados a objetos, que también proveen de una buena consistencia y acceso a los datos.

V. REFERENCIAS BIBLIOGRÁFICAS

[1] Riak , Anonimo (S.F), Disponible en: <https://en.wikipedia.org/wiki/Riak>.

[2] Riak: Sistema NoSQLpara todos los gustos, Master Big Data Universidad Alcala (29 febrero, 2018) , Disponible en: <https://www.master-bigdata.com/riak-sistemas-nosql-todos-los-gustos/>

[3] ¿Que es una base de datos NoSQL?, Andres Araujo (19 Abril, 2016), Disponible en: <https://blogs.oracle.com/spain/qu-es-una-base-de-datos-nosql>.

VI. ANEXOS

1. Riak no es (Seleccione uno o mas de una):

1. Un gestor de base de datos SQL

2. Una base de datos clave-valor

3. Un gestor de base de datos NO-SQL

4. Una base de datos relacional

2. Que comando se utiliza para saber si riak esta en funcionamiento:

a) sudo riak start

b) sudo riak ping

c) sudo riak function

d) sudo riak node.function()

3. ¿Riak tiene un entorno de desarrollo equivalente al de SQL DEVELOPER?

a) si

b) no

4. ¿Por medio de qué concepto maneja los datos riak?

a) buckets

b) tables

c) node

d) Trees

5. ¿Cuáles son las caracterisiticas principales al trabajar con no-sql?

1. Manejo optimizado de poca información

2. Manejo de grandes cantidades de información

3. Escalabilidad lineal y predecible

4. Performance en la base de datos

6. En Riak ¿qué significa CRUD?

a) CAUTION, READ, UPLOAD, DEF

B) CREATE, RESTORE, UPDATE, DELETE

C) CREATE, READ, UPDATE, DELETE

D) CREATE, RESTORE, UPLOAD, DELETE

7. ¿Qué empresas utilizan Riak como base de datos?

a) McDonalds, Facebook, Google

b) Amazon,Riot Games,Best Buy

c) Best Buy,Riot Games, Github

d) Github,Best Buy,Amazon

8. ¿Cuál es la empresa que desarrolla Riak?

a) Oracle

b) Riak Enterprisse

c) Basho

d) Apache

9. ¿Python es el unico lenguaje en el que puedes desarrollar con Riak?

a) Falso

b) Verdadero

10. ¿RiakTS es una version de Riak SQL?

a) Verdadero

b) Falso