**Base De Datos De Modelos Múltiples**

La reproduccion de bases de datos NoSQL es una respuesta a las necesidades de las aplicaciones modernas. Aún así, no todos los datos pueden ser ajustados a un modelo NoSQL particular, razón por la cual existen tantas opciones de bases de datos diferentes en el mercado. Como resultado, las organizaciones ahora se enfrentan a una gran cantidad de bases de datos dentro de su infraestructura.

Los beneficios para las organizaciónes son amplios, pero algunos de los beneficios más importantes incluyen:

**1. Consolidación**

En el espacio NoSQL en particular, los ingenieros se enfrentan a muchas opciones cuando deciden cómo modelar y almacenar datos.

**2. Escalado De Rendimiento**

A medida que crece el uso de una aplicación, la necesidad de rendimiento de la base de datos también aumenta.

**3. Complejidad Operacional**

Los entornos fragmentados causados ​​por ejecutar diferentes bases de datos aumentan la complejidad tanto de las operaciones como del desarrollo. El objetivo de la persistencia políglota es utilizar el mejor componente para el trabajo, pero en la práctica significa que puede terminar con múltiples bases de datos, cada una con sus propios requisitos operativos y de almacenamiento

**4. Flexibilidad**

A menudo es incómodo e ineficaz meter muchos datos en un solo modelo de datos. Un enfoque multimodelo implica mapear múltiples modelos de datos en un único motor de almacenamiento subyacente que puede admitir diferentes casos de uso y aplicaciones.

**5. Confiabilidad**

La confiabilidad de la base de datos también es un problema cuando se ejecutan múltiples bases de datos, ya que cada sistema de base de datos podría ser un único punto de falla para el sistema y la aplicación de mayor tamaño.

**6. Consistencia De Los Datos**

Sin la funcionalidad de transacciones de alto nivel integrada en su aplicación, no hay soporte para transacciones en diferentes sistemas de bases de datos. En consecuencia, no hay una buena manera de mantener la coherencia entre los diferentes modelos. Supongamos que su aplicación recibe una secuencia de datos sobre la actividad del usuario y decide almacenar elementos de datos relacionados en una estructura de series de tiempo, formato de gráfico y formato de documento.

**7. Tolerancia A Fallas**

Asegurar que un sistema con muchos componentes de cualquier tipo sea tolerante a fallas no es una tarea fácil, por decir lo menos. E integrar múltiples sistemas que fueron diseñados para ejecutarse de forma independiente, de modo que proporcionan tolerancia a fallas en todo el sistema en conjunto, impone costos de ingeniería y operativos significativos. Desafortunadamente, las implementaciones de sistemas heterogéneos requieren que su equipo tenga experiencia con cada componente para que el sistema en general funcione correctamente. Debido a que cada sistema es diferente y tiene diferentes requisitos, sin embargo, este enfoque lleva mucho tiempo y es costoso. E incluso entonces, la tolerancia a fallas de todo su sistema depende entonces del subsistema más débil en el back-end.

**8. Costo**

Al usar más, los distintos sistemas de bases de datos aumentan los costos según el hardware, el software y las necesidades operativas asociadas con cada sistema.

**9. Transacciones**

Los sistemas de bases de datos relacionales, generalmente implementados en una sola máquina, generalmente ofrecen fuertes garantías transaccionales para las operaciones de la base de datos, lo que permitió a las aplicaciones y desarrolladores de aplicaciones comprender con certeza el estado actual de la base de datos en un momento determinado. Sin embargo, es un desafío proporcionar transacciones entre varias máquinas, y casi todas las bases de datos NoSQL no ofrecen garantías transaccionales debido a sus diseños arquitectónicos.

**10. Mejores Aplicaciones**

Intentar ejecutar diferentes bases de datos para alimentar una aplicación puede ser una pesadilla operacional y de desarrollo. Por el contrario, una aplicación que es compatible con una base de datos de modelos múltiples obtiene los beneficios de escalabilidad, tolerancia a fallas y en un sistema bien diseñado, alto rendimiento integrado en el producto.