Comparación de Despliegue de un Gestor de Base de Datos NoSQL Mediante Docker

Marko Antonio Rivas Rios
Jorge Luis Mamani Maquera
Orlando Antonio Acosta Ortiz
Yofer Nain Catari Cabrera
Orestes Ramirez Ticona
Roberto Zegarra Reyes
Universidad Privada de Tacna

Junio 22, 2019

Abstract

Docker is an open source project created in 2013 and which has been a revolution for the development and deployment of operations. Docker abstracts the host's hardware and operating system by running the applications in containers, isolated compartments that contain all the resources for an application or service. In this work we will see how to use Docker for the development of simple applications, learning how to deploy a NOSQL database with Docker.

Abstract

Docker es un proyecto open source creado en 2013 y que ha supuesto una revolución para el desarrollo y despliegue de operaciones. Docker abstrae el hardware y el sistema operativo del host ejecutando las aplicaciones en contenedores, compartimentos aislados que contienen todos los recursos para una aplicación o servicio.

En este trabajo veremos cómo usar Docker para el desarrollo de aplicaciones sencillas, aprendiendo a desplegar una base de datos NOSQL con Docker.

I. Introducción

Eservicios monolíticos están dando paso lentamente al enjambre microservicios pequeños y autónomos que trabajan en conjunto. El proceso va acompañado de otra tendencia del mercado: la contenerización. Juntos, ayudan a construir sistemas sin precedentes. La contenerización cambia no sólo la arquitectura de los servicios, sino también la estructura de ambientes utilizados para crearlos.

Ahora, cuando el software se distribuye en contenedores, los desarrolladores tienen plena libertad para decidir qué aplicaciones necesitan. Como resultado, incluso los entornos complejos, como los servidores de grades bases de datos e infraestructura de análisis complejos pueden crear instancias en cuestión de segundos. El desarrollo de software se hace más fácil y más eficaz.

П.	MΑ	TERIA	LES Y	Мéт	ODOS

i. Creación de una base de datos NoSQL en docker

ii. Inserción de datos y Consulta de datos (en una base de datos NOSQL)

iii. Comparacion de distintos tipos de base de datos NoSQL

Tabla comparativa					
	MongoDB	Cassandra	DynamoDB		
Descripción	Servicio de base de	Almacén de gran	Una de las tiendas		
	datos alojado y	columna basado en	de documentos más		
	escalable de Amazon	ideas de BigTable y	populares		
	con los datos	DynamoDB			
	almacenados en la	•			
	nube de Amazonas				
Modelo de base de	Document store: Las	Wide column store:	Document store: Las		
datos primaria	tiendas de	Las tiendas de	tiendas de		
·	documentos,	columnas anchas,	documentos,		
	también Ilamadas	también llamadas	también Ilamadas		
	sistemas de bases de	tiendas de discos	sistemas de bases de		
	datos orientadas a	extensibles,	datos orientadas a		
	documentos, se	almacenan datos en	documentos, se		
	caracterizan por su	registros con la	caracterizan por su		
	organización de	capacidad de	organización de		
	datos sin esquema.	mantener un gran	datos sin esquema.		
	Key-value store: Las	número de columnas			
	tiendas de valores	dinámicas.			
	clave son				
	probablemente la				
	forma más simple de				
	sistemas de				
	administración de				
	bases de datos. Solo				
	pueden almacenar				
	pares de claves y				
	valores, así como				
	también recuperar				
	valores cuando se				
	conoce una clave.				
Versión inicial	2012	2008	2009		
Basado en la nube	sí	no	no		
Lenguaje de		Java	C++		
implementación					
Esquema de datos	sin esquema	sin esquema	sin esquema		
Índices secundarios	sí	restringido	sí		
SQL	no	Sentencias DML y	no		
		DDL similares a SQL			
		(CQL)			
API y otros métodos	API RESTful HTTP	Protocolo	protocolo		
de acceso		propietario	propietario usando		
		Ahorro	JSON		
Lenguajes de	.Net	C#	С		
programación	Erlang	C++	C#		
admitidos	Groovy	Clojure	C++		
	Java	Erlang	Clojure info		
	JavaScript	Go	D info		
	Perl	Haskell	Erlang		
	PHP	Java	Go info		
	1 1 11	7G V G	GO IIIIO		

	T =		
	Python Ruby	JavaScript info Perl	Groovy info Haskell
		PHP	Java
		Python	JavaScript
		Ruby	PHP
		Scala	PowerShell info
		Scala	Prolog info
			Python
			_
0 :			Ruby
Scripts del lado del	no	no	JavaScript
servidor	-1	_f	
Triggers Métodos de	SÍ	SÍ	no Chandina
	Sharding	Sharding	Sharding
particionamiento		6tdl'!/	D!'!'t
Métodos de	sí	factor de replicación	Replicación maestro-
replicación		seleccionable	esclavo
Llaves Foraneas	no	no	no
Conceptos de	no	no	no
transacción		,	,
Concurrencia	sí	sí	sí
Durabilidad	sí	sí	sí
Entornos de	hosting de amazon	BSD	Linux
operación		Linux	OS X
		OS X	Solaris
		Windows	Windows
Ventajas	Totalmente	Ningún punto único	MongoDB mantiene
Ventajas	Totalmente administrado	Ningún punto único de falla garantiza el	MongoDB mantiene las características
Ventajas			
Ventajas		de falla garantiza el	las características
Ventajas	administrado	de falla garantiza el 100% de	las características más valiosas de las
Ventajas	administrado Amazon DynamoDB	de falla garantiza el 100% de	las características más valiosas de las bases de datos
Ventajas	administrado Amazon DynamoDB es un servicio de base	de falla garantiza el 100% de disponibilidad.	las características más valiosas de las bases de datos relacionales: coherencia fuerte,
Ventajas	Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente	de falla garantiza el 100% de disponibilidad. Simplicidad	las características más valiosas de las bases de datos relacionales:
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta
Ventajas	Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices
Ventajas	Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad.	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL.
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL.
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL.
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL. MongoDB proporciona flexibilidad de
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL. MongoDB proporciona flexibilidad de modelo de datos,
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL. MongoDB proporciona flexibilidad de modelo de datos, escalabilidad elástica
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL. MongoDB proporciona flexibilidad de modelo de datos, escalabilidad elástica y alto rendimiento y
Ventajas	administrado Amazon DynamoDB es un servicio de base de datos NoSQL en la nube totalmente administrado; usted simplemente crea una tabla de base de datos, establece su rendimiento y deja que el servicio se	de falla garantiza el 100% de disponibilidad. Simplicidad operativa para el menor costo total de propiedad. La mejor escalabilidad de las	las características más valiosas de las bases de datos relacionales: coherencia fuerte, lenguaje de consulta expresivo e índices secundarios. Como resultado, los desarrolladores pueden construir aplicaciones altamente funcionales más rápido que las bases de datos NoSQL. MongoDB proporciona flexibilidad de modelo de datos, escalabilidad elástica

Desventajas - DynamoDB es una base de datos NoSQL Eso significa que no puede hacer consultas complejas o de unión en escribir objetos grandes (BLOB) en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. La desventaja más grande para Cassandra es que no exormatible con los índices B-Tree y, por lo tanto, no diseña como si nunca hubieran sido una opción. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transacciones, excepto en el nivel de registro. Si desea una transacción en un solo registro. Si desea una transacción en un sol		ı		
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoBB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. La desventaja más paplicaciones y entregarlas a escala casi ilimitada en hardware básico. No un ustilor de las diseña como sí nunca hubieran sido una opción. Se escala bien en un rango estrecho, pero otras soluciones nos nun sustituto del as bases relacionales. Los contadores nos nuny confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Sin transacciones, excepto en el nivel de registro. Si desea una transacción en un solo registro. Difficil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				NoSQL. Como
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajustes S3 es o tra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. - DynamoDB es una base de datos grande para de mos de compatible con los índices B-Tree y, por lo tanto, no admite consultas de rango. - No se pueden escribir objetos grandes (BLOB) en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajustes S3 es o tra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Sin transacciones, excepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantia de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				· ·
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoBB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBs en dynamo BB. Definitivamente puede almacenar metadatos para BLOBs en dynamoBB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. La desventaja más grande para (Cassandra es que no a compatible con los índices B-Tree y, por lo tanto, no amite consultas de rango. -Los -Los -CounterColumns no son un sustituto del as bases relacionales. Los contadores no son un sustituto de la derementos, este de no ser el resultado deseado. Si está haciendo decrementos, este presultado deseado. Si está haciendo decrementos, este presultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y es emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantia de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas compleis a o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamo DB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. - DynamoDB es una base de datos NoSQL. Eso significa que hardares para base de datos mon del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. - DynamoDB es una base de datos grande para (as a sciulifizad en hardware básico. La desventaja más grande para (as a scue no tacami la desventaja más grande para (as sumano para de ja para escalar. La desventaja más grande para (as a scue no hardware básico. La desventaja más grande para (as a scue no hardware básico. No utiliza joins. Se diseña como si nunca hubieran sido una opción. Se escala bien en un arago estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transacción, de lese ajuna transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contra con una licencia Entreprise. No hay parches, tiene que hacer actualizaciones completas y las actualizacione				*
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamoDB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. La desventaja más grande para (Cassandra es que no es compatible con los índices B-Tree y, por lo tanto, no admite consultas de rango. Los CounterColumns no son un sustituto del subsaes relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Sin transaccione, Sin transacción en en solo registro. Sin transaccione, sexepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Deficil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamo DB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el euso de orden lexicográfico, el uso el uso de orden lexicográfico, pel uso flectro y la paralelización del os GET. La desventaja más grande para Cassandra es que no es compatible con los fides B-Tree y, por lo tanto, no admite consultas de rango. Se escala bien en un admite consultas de rango. Se escala bien en un rango estrecho, pero otras soluciones no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Sin transaccione, excepto en el nivel deramento de las becerementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones compl				
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en CS3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. La desventaja más grande para Casandra es que no ser el ros damíte consultas de rango. No Moutiliza joins. Se diseña como si diseña como si diseña como si diseña como si una opción. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transacción en un solo registro. Si desea ucontar con una licencia Enterprise. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y existe en la rededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				_
Desventajas - DynamoDB es una base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamoDB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en CS3. La recuperación del ajuste S2 as otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. - Da disciplinar de disciplinar de la dato deseado. La desventaja más grande para (Cassandra es que no es compatible con los índices B-Tree y, por lo tanto, no admite consultas de rango. - Los CounterColumns no son un sustituto del autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y la				casi ilimitada en
base de datos NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. grande para Cassandra es que no es compatible con los índices B-Tree y, por lo tanto, no admite consultas de rango. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transacciones, excepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y excepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches invano poción.				hardware básico.
NoSQL. Eso significa que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Cassandra es que no es compatible con una opción. Cassandra es que no es compatible con una opción. Cassandra es que no es compatible con una opción. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transaccione, execepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difficil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones el controlador dado, por lo que las pruebas deben programare antes	Desventajas	- DynamoDB es una	La desventaja más	-
que no puede hacer consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. es compatible con los índices B-Tree y, por lo tanto, no admite consultas de rango. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Si ergistro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		base de datos	grande para	diseña como si
consultas complejas o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Ios (CounterColumns no son un sustituto del autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difficil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas de que su actualizaciones completas deben programarse antes		NoSQL. Eso significa	Cassandra es que no	nunca hubieran sido
o de unión en dynamoDB. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Se escala bien en un rango estrecho, pero otras soluciones NoSQL son mejores para escalar. CounterColumns no son un sutituto del autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y em construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		que no puede hacer		una opción.
dynamodb. Además, no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamodb. Definitivamente puede almacenar metadatos para BLOBS en dynamodb, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. admite consultas de rango. rango estrecho, pero otras soluciones NoSQL son mejores para escalar. Sin transacciones, excepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Sio está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		consultas complejas	los índices B-Tree y,	
no ACID, ya que no es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. rango. rango. corras soluciones NoSQL son mejores para escalar. Sin transacciones, excepto en el nivel de registro. Si desea una transacción de la para contar con un solo registro. puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		o de unión en	por lo tanto, no	Se escala bien en un
es un RDBMS - No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y el sa actualizaciones completas y el sa actualizaciones completas y el miter varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		dynamoDB. Además,	admite consultas de	rango estrecho, pero
- No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. - Los CounterColumns no son un sustituto del autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		no ACID, ya que no	rango.	otras soluciones
- No se pueden escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. CounterColumns no son un sustituto del autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		es un RDBMS		NoSQL son mejores
escribir objetos grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Sin transacciones, excepto en el nivel de registro. Si desea una transacción, debe ajustar toda la transacción en un solo registro. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes			-Los	para escalar.
grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		- No se pueden	CounterColumns no	
grandes (BLOB) en dynamo DB. Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. autoincrement de las bases relacionales. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		escribir objetos	son un sustituto del	Sin transacciones,
Definitivamente puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Los contadores no son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		grandes (BLOB) en	autoincrement de las	excepto en el nivel
puede almacenar metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. son muy confiables. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		dynamo DB.	bases relacionales.	de registro. Si desea
metadatos para BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso deatorio y la paralelización de los GET. Si está haciendo decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		Definitivamente	Los contadores no	una transacción,
BLOBS en dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. decrementos, este puede no ser el resultado deseado. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		puede almacenar	son muy confiables.	debe ajustar toda la
dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		metadatos para	Si está haciendo	transacción en un
dynamoDB, pero objetos reales que puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Difícil de asegurar correctamente sin contar con una licencia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas y las actualizaciones completas y emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		BLOBS en	decrementos, este	solo registro.
puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Serior de los de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		dynamoDB, pero	puede no ser el	
puede almacenar en S3. La recuperación del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Serior de los de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		objetos reales que	resultado deseado.	Difícil de asegurar
del ajuste S3 es otra cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Bichicomia Enterprise. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		puede almacenar en		correctamente sin
cosa que debe tener en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Roberto No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		S3. La recuperación		contar con una
en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		del ajuste S3 es otra		licencia Enterprise.
en cuenta para obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. No hay parches, tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		cosa que debe tener		
obtener un mejor rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. tiene que hacer actualizaciones completas y las actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				No hay parches,
rendimiento, como el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
el uso de orden lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. completas y las actualizaciones construitae construitae construitae con		rendimiento, como		actualizaciones
lexicográfico, el uso el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. actualizaciones completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				completas y las
el uso de orden lexicográfico, el uso aleatorio y la paralelización de los GET. Completas y las actualizaciones completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		lexicográfico, el uso		
lexicográfico, el uso aleatorio y la paralelización de los GET. GET. CENTONCES tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				completas y las
aleatorio y la paralelización de los GET. Completas se emiten varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		lexicográfico, el uso		
paralelización de los GET. varias veces al año. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
GET. Entonces tienes que construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
construir un programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes		l ·		
programa de parches alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
alrededor de ellos. No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
No hay garantía de que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				' - '
que su actualización funcione con su controlador dado, por lo que las pruebas deben programarse antes				
funcione con su controlador dado, por lo que las pruebas deben programarse antes				
controlador dado, por lo que las pruebas deben programarse antes				-
por lo que las pruebas deben programarse antes				
pruebas deben programarse antes				
programarse antes				
l de la actualización				de la actualización.

CouchDB:

web.

CouchDB es catalogado muchas veces como una base de datos NoSQL, un término que se hizo cada vez más popular a finales de 2009 y principios de 2010. Si bien este término es una caracterización más bien genérica de una base de datos, o almacén de datos, sí define claramente un descanso de SQL tradicional bases de datos. Una base de datos CouchDB carece de un esquema o estructuras de datos pre-definidos rígidos tales como tablas. Los datos almacenados en CouchDB es un documento (s) JSON. La estructura de los datos, o documento(s), puede cambiar dinámicamente para adaptarse a las necesidades cambiantes. CouchDB es una base de datos que abarca por completo la web. Almacene sus datos con documentos JSON. Tenga acceso a sus documentos y consultar sus índices con su navegador web, a través de HTTP.Indice, combinar y transformar sus documentos con JavaScript. CouchDB funciona bien con la web moderna y aplicaciones móviles. Usted puede incluso servir aplicaciones web directamente de CouchDB. Y usted puede distribuir sus datos o sus aplicaciones, de manera eficiente mediante la replicación incremental de los CouchDB. CouchDB soporta configuraciones maestro-maestro con detección automática de conflictos. CouchDB viene con una serie de características, como la transformación de documentos sobre la marcha y notificaciones de cambio en tiempo real, que hace que el desarrollo de

Las principales características son las siguientes:

• Almacenamiento de documentos:

Almacena los datos como documentos esto es, uno o más pares campo/valor expresados en JSON. Los valores de los campos pueden ser datos simples como cadenas de caracteres, números o fechas. Pero también se pueden usar listas ordenadas y vectores asociativos. Todos los documentos en una base de datos CouchDB tienen un identificador único y no requieren un esquema determinado.

aplicaciones web una brisa. Incluso viene con un fácil utilizar la consola de administración

• Vistas e índices Map/Reduce:

Los datos almacenados se estructuran por medio de vistas. En CouchDB, cada vista se construye por medio de una función JavaScript que actúa como la mitad Map de una operación map/reduce. La función recibe un documento y lo transforma en un único valor, retornándolo. CouchDB puede indexar vistas y mantener actualizados esos índices a medida que se agregan, eliminan o actualizan documentos.

• Arquitectura distribuida con replicación:

Se diseñó con teniendo en mente la replicación bidireccional (o sincronización) y la operación off-line. Eso significa que múltiples réplicas pueden tener cada una sus propias copias de los mismos datos, modificarlas y luego sincronizar esos cambios en un momento posterior.

• Interfaz REST:

Todos los ítems tienen una URI única que queda expuesta vía HTTP. REST usa los métodos HTTP POST, GET, PUT y DELETE para las cuatro operaciones básicas CRUD (Create, Read, Update, Delete) con todos los recursos.

• Consistencia Eventual:

Garantiza consistencia eventual para poder ofrecer tanto disponibilidad como tolerancia a las particiones.

• Hecha para operar offline:

Puede replicar datos a dispositivos (como smartphones) que pueden quedar offline y manejar automáticamente la sincronización de los datos cuando el dispositivo vuelve a estar en línea.

Neo4j:

Es una base de datos orientada a grafos escrita en Java, es decir la información se almacena de forma relacionada formando un grafo dirigido entre los nodos y las relaciones entre ellos. Se integra perfectamente con múltiples lenguajes como Java, PHP, Ruby, .Net, Python, Node, Scala, etc. La base de datos está embebida en un servidor Jetty. Está especialmente indicada para modelar redes sociales y sistemas de recomendación.

Se distribuye en dos versiones: la community edition (open source) y la Enterprise edition. Para hacer pruebas de concepto nos basta con la community edition pero si quieres sacarle todo el partido a Neo4j la opción enterprise es la más recomendable ya que permite ponerla en cluster, monitorización, backups en caliente y un sistema de cache de alto rendimiento, además de soporte de sus creadores.

Otra de las ventajas que tiene Neo4j es que se pueden efectuar las consultas directamente a través de un API Rest lo que hace especialmente interesante su integración con aplicaciones web.

Principales características de neo4j:

- Alto desempeño y alta disponibilidad (Escalamiento de lectura) Soporte sólido y real para transacciones ACID.
- Escalable: 32 miles de millones de Nodos, 32 miles de millones de Relaciones, 64 miles de millones de Propiedades.
- Servidor con una API REST o usable como una biblioteca Java.

III. RESULTADOS

Comparaciones de 2 Bases de Datos NoSQL

- i. Grafos
- ii. Tabular (Column-Store)

iii. Documental

iv. Clave-Valor

IV. Discusión

V. Conclusiones

REFERENCES

- 1. http://revistatelematica.cujae.edu.cu/ index.php/tele/article/view/23/21
- 2. https://programarfacil.com/blog/ que-es-un-orm/
- 3. https://www.beeva.com/beeva-view/tecnologia/mas-alla-de-la-virtualizacion-contenedores/
- 4. https://searchdatacenter.techtarget.com/ es/definicion/virtualizacion-basada-en-contenedores-virtualizacion-a-nivel-de-sistema-operativo
- 5. https://www.incibe-cert.es/blog/asegurando-virtualizacion-tus-sistemas-control
- 6. http://www.datakeeper.es/?p=716

7.https://sigmodrecord.org/publications/sigmodRecord/1012/pdfs/04.surveys.cattell.pdf

8.https://ieeexplore.ieee.org/abstract/document/6625441

9.-http://nosql-database.org/