Cuaderno Activa
Revista científica de la Facultad de Ingeniería

Rendimiento de tecnologías NoSQL sobre cantidades masivas de datos.

Performance of NoSQL Technology on Massive Amounts of Data.

María Florencia Pollo-Cattaneo*
Marcelo López Nocera**
Giovanni Daián Rottoli***

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

Tipo de artículo: Resultado de Investigación

Recibido: 25 de agosto, 2014
Aceptado: 30 de octubre, 2014

Resumen

Debido al gran crecimiento de Internet de los últimos años y la llegada del fenómeno de Big Data, surgen nuevas cuestiones a ser consideradas a la hora de almacenar y consultar cantidades masivas de datos que, en general, las bases de datos relacionales tradicionales no pueden abarcar. Estas cuestiones incluyen desde la capacidad de distribuir y escalar el procesamiento o el almacenamiento físico, hasta la posibilidad de utilizar esquemas o tipos de datos no usuales. Las bases de datos NoSQL han surgido para dar respuesta a estas problemáticas mediante la utilización de nuevos enfoques, los cuales pueden diferir enormemente unos de otros de acuerdo al tipo de problemas que buscan solucionar. El presente trabajo detalla las características de estas nuevas tecnologías y realiza un estudio comparativo sobre el rendimiento de carga masiva y consulta de distintos motores de bases de datos NoSQL, con respecto a diversos tipos de datos para encontrar aquellas tecnologías que se adaptan mejor a las distintas características de estas estructuras.

Palabras clave: bases de datos NoSQL, carga masiva de datos, Big Data.

Abstract

Due to the enormous growth of the Internet in the last years and to the arrival of the Big Data phenomenon, new aspects must be considered when storing and consulting massive amounts of data which traditional relational databases cannot generally cover. These aspects include requirements ranging from the ability to distribute and scale the processing or the physical storage to the possibility of using unusual data structures and schemes. NoSQL databases have been created to answer these problems using new approaches that can be very different between them according to the types of problems to solve. This article details the characteristics of these new technologies and performs a comparative study on the massive loading and query performance of different NoSQL database engines regarding different types of data in order to find the technologies best suited to the various characteristics of these structures.

Keywords: NoSQL databases, massive data loading, Big Data.

*Magister en Ingeniería en Software. Profesor Titular Concursado. Universidad Tecnológica Nacional, Argentina. flo.pollo@gmail.com

**Magister en Ingeniería en Sistemas de Información. Profesor. Universidad Tecnológica Nacional, Argentina. zappapet@yahoo.com

***Analista Universitario de Sistemas. Estudiante carrera Ingeniería en Sistemas de Información. Universidad Tecnológica Nacional, Argentina. gd.rottoli@gmail.com

Introducción

Las bases de datos SQL parecen no ser suficientes para tratar con tan amplio universo de problemas que vienen ligados a la aparición de Big Data. Esto generó la llegada de las tecnologías para el almacenamiento de datos englobadas dentro del concepto de NoSQL (Sadalage et al., 2013).

Estas herramientas sacrifican usualmente algunas cuestiones relacionadas con la atomicidad, consistencia, aislamiento y la durabilidad, para obtener ventajas como el rendimiento, la posibilidad de escalar el almacenamiento físico, distintas formas de representación de los datos, entre otros. No todas las tecnologías NoSQL son idóneas para tratar con todo tipo de estructuras de datos, por lo que es preciso saber optar por cada una de ellas a la hora de implementar un sistema de bases de datos no tradicional, en función a los datos y las relaciones entre los mismos (Strauch et al., 2011).

El presente artículo evalúa las características de carga de datos y consultas de los mismos, tanto en bases de datos tradicionales -las cuales utilizan lenguaje de consulta SQL-, como en bases de datos NoSQL, frente a distintas estructuras y tipos de datos, y propone una alternativa para realizar un diseño de una arquitectura de persistencia adaptada a las distintas necesidades.

El término NoSQL comienza a ser utilizado mayormente a partir del año 2009, abarcando todas aquellas tecnologías de bases de datos que no utilizan el lenguaje ANSI SQL estándar para la escritura de sus consultas (Sadalage et al., 2013). NoSQL es más un movimiento, una nueva tendencia, que una tecnología (Ramírez et al., 2013).

Por lo general, estas tecnologías están preparadas para correr en sistemas con arquitecturas de procesamiento y almacenamiento distribuido –lo que las hacen fácilmente escalables (López, 2012; Del Busto et al., 2013)- y siguen modelos de datos distintos al modelo relacional tradicional.

Por lo general, además, operan sin esquemas, por lo que no es necesario que los registros de la base de datos sean iguales, sin tener que redefinir la estructura (Bugiotti et al., 2013), pudiendo además poseer entradas que contengan a su vez otras estructuras no tradicionales (Hecht, 2011). Estas características hacen que NoSQL sea idónea para trabajar con grandes cantidades de datos y a su vez tan variados, peculiaridades de Big Data (Strauch et al., 2011).

Metodología

El método de investigación utilizado pretende resolver el siguiente interrogante: ¿Cuál es el impacto de la naturaleza de los datos sobre la performance de los distintos tipos de tecnologías NoSQL? Y para resolverlo se propone la ejecución de pruebas de carga masiva y consulta de datos sobre distintos tipos de tecnologías SQL y NoSQL, a partir de un rastreo bibliográfico de algunas bases de datos no tradicionales.

Bases de datos no tradicionales

Bajo el título de NoSQL, como anteriormente se menciona, existen distintas clasificaciones, siendo la principal aquella centrada en el modelo de datos con el que trabajan. Según este criterio, existen cinco grandes tipos de bases de datos no tradicionales (Nayak et al., 2013), más el presente artículo solo se centra en cuatro de ellos:

Bases de datos clave-valor

Los datos almacenados consisten en dos partes: una que representa la clave, y el dato en sí, que se referencia como el valor en el par clave-valor. Las claves son usadas como índices para las búsquedas, pudiendo estos ser compuestos formados por criterios distintos, como se observa en la Figura 1.

Estas bases de datos sacrifican la consistencia de los datos para obtener escalabilidad. Como punto débil, la falta de un esquema hace mucho más difícil interpretar los datos. Ejemplos de estas bases de datos son Redis, RIAK y Amazon DynamoDB (Nayak et al., 2013)

Clave Compuesta	Valor
Legajo: Dpto: Sucursal	Nombre

Figura 1. Estructura de una entrada de una base de datos clave-valor
Fuente: Elaboración propia (2014).

Bases de datos orientadas a columnas o column-family

Son una evolución de las bases de datos clave-valor. En estas bases de datos, cada clave está asociada con uno o más atributos (columnas), como se observa en la Figura 2, de manera que puedan ser accedidos más rápidamente. Esto posee las ventajas de la posibilidad de escalar, brinda un esquema más rígido que las bases de datos clave-valor, aun sacrificando la consistencia. Estas tecnologías son apropiadas para la minería de datos. Ejemplos de bases de datos orientadas a columnas son Cassandra y Big Table (Nayak et al., 2013)

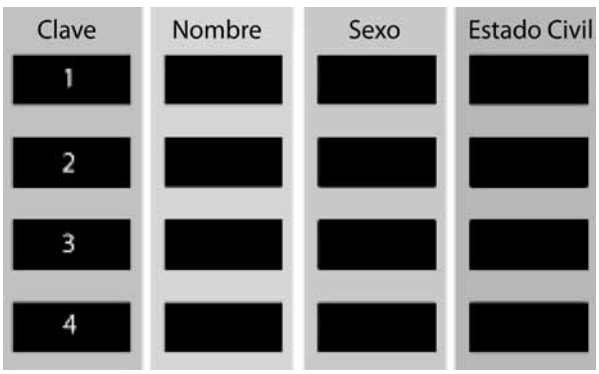


Figura 2. Estructura de una base de datos orientada a columnas
Fuente: Elaboración propia (2014).

Bases de datos documentales o basadas en documentos

Almacenan los datos en forma de documentos. Dentro de estos documentos es posible anidar otros documentos relacionados, como si se tratase de carpetas físicas, como se puede ver en la Figura 3. En sí, los documentos son muy similares a los registros en las bases de datos tradicionales, pero mucho más flexibles por su falta de esquemas. Los documentos son accesibles por medio de claves únicas y la información que almacenan no tiene que estar normalizada. Los blogs usualmente utilizan estas tecnologías, entre las cuales se pueden mencionar MongoDB y CouchDB (Nayak et al., 2013)

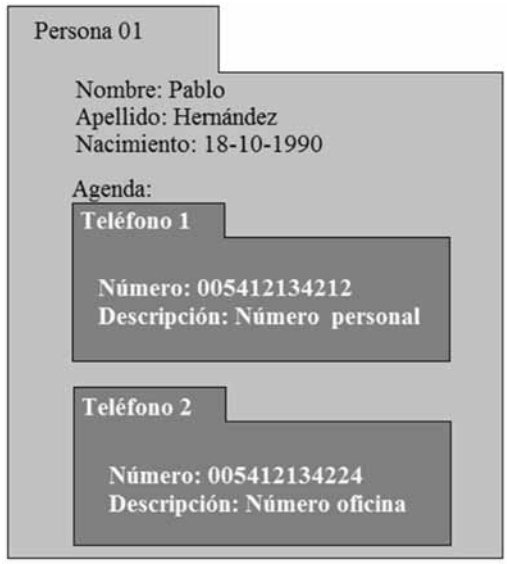


Figura 3. Ejemplo de representación de una persona en una base de datos documental
Fuente: Elaboración propia (2014).

Bases de datos gráficas o basadas en grafos

En estas bases de datos se almacenan las entradas en forma de nodos y relaciones entre ellos. Son muy eficientes para manejar este tipo de relaciones, ya que utilizan algoritmos basados en la teoría de grafos para ejecutar las consultas. Los datos almacenados son libres de esquemas, por lo que cada nodo y cada relación pueden poseer atributos distintos; un ejemplo de esto se observa en la Figura 4.

Estas tecnologías son utilizadas para una amplia variedad de aplicaciones de redes sociales, sistemas de recomendación, manejo de contenido, entre otras, siendo una de las más utilizadas Neo4J. (Nayak et al., 2013)

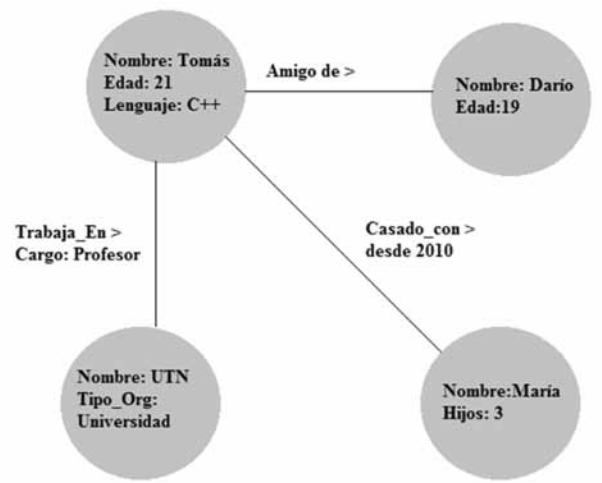


Figura 4. Ejemplo de relaciones en una base de datos de grafos
Fuente: Elaboración propia (2014).

Usualmente, a la hora de optar por una tecnología NoSQL, se intenta utilizar la que mejor se adapte a la naturaleza de los datos que queremos modelar. Por ejemplo, lo ideal para almacenar relaciones es una base de datos gráfica (Hecht, 2011).

Resultados

Para evaluar la performance de las distintas tecnologías NoSQL se propuso un modelo de datos, en principio relacional y fácilmente adaptable a las distintas tecnologías, sobre el cual se generaron datos propios para su carga masiva y posterior consulta en las distintas bases de datos con las que se cuentan. La estructura utilizada se muestra en la Figura 5.

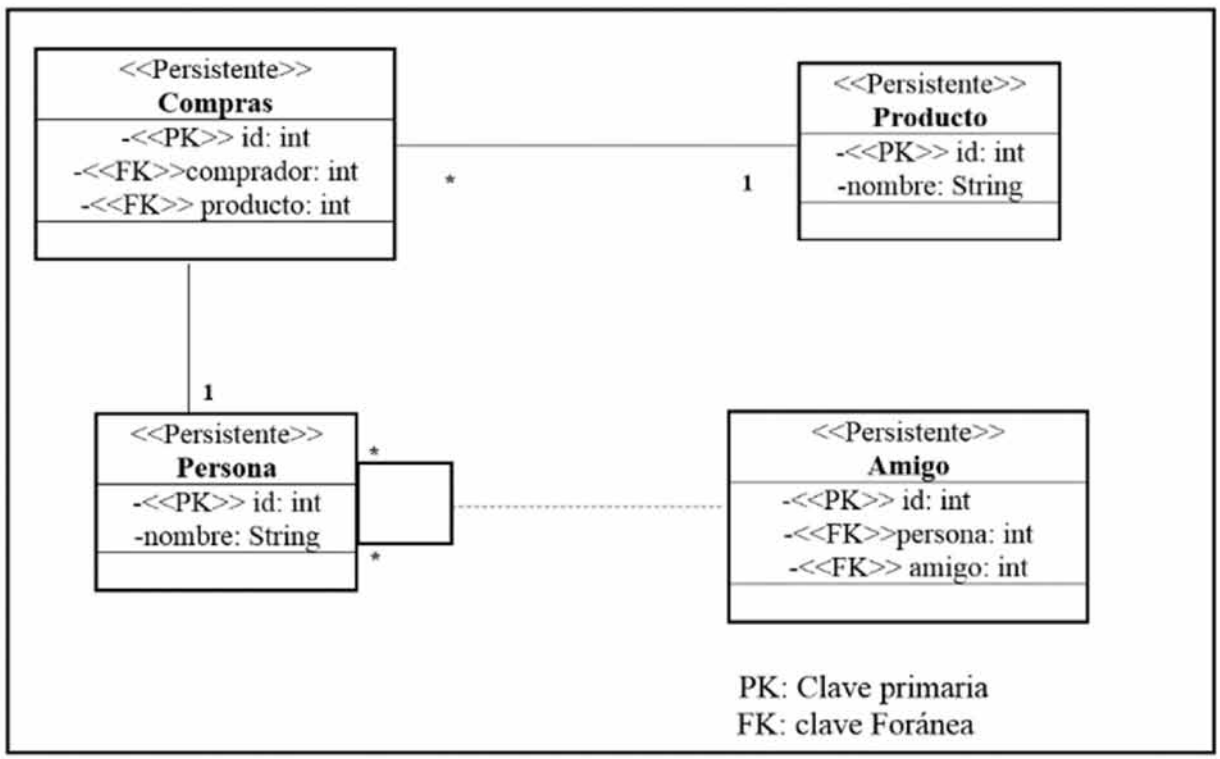


Figura 5. Estructura de los datos para pruebas
Fuente: Elaboración propia (2014).

La cantidad de datos correspondientes de cada entidad fue de 1.000.000 (un millón) para las entidades "Persona", "Compras" y "Amigos" y de 100 (cien) para los productos. Todos estos datos se grabaron sobre archivos CSV para su fácil y uniforme acceso (Róttoli, 2014a).

Las pruebas se realizaron sobre un único nodo, consistente en una PC de escritorio con 4 Gb de memoria RAM, procesador Intel Core I5 @ 2.30 GHz y disco rígido de 500 Gb, sobre un sistema operativo Ubuntu 14.04.

Posteriormente, se prosiguió con la carga masiva de los datos en las distintas tecnologías, optando por PostgreSQL como base de datos relacional, Redis como base de datos clave-valor, Cassandra para el modelo orientado a columnas, MongoDB para documentos y Neo4J como motor basado en grafos.

Todas estas alternativas se eligieron por su gran utilización y por ser herramientas Open Source (políticas generales de la institución educativa promueven la utilización de este tipo de herramientas). Por cada una de las tecnologías se cargaron los datos, midiendo los tiempos de ejecución de cada una de las consultas, utilizando scritps en Python 2.7 (Róttoli, 2014b), realizando sobre las mismas, posteriormente, consultas complejas que relacionen todas las entidades, por ejemplo, devolver los productos comprados por los amigos de una persona. Los resultados obtenidos se detallan en la Tabla 1.

Modelo	Base de datos	Personas	Productos	Compras	Amigos	Consulta
Relacional	PostgresSQL	**11.617**	**0.011**	84.019	44.256	0.015
Clave-Valor	Redis	59.5927	0.055	77.301	63.890	**0.00059**
Documental	MongoBD	28.195	0.014	26.269	29.659	0.823
Columnas	Cassandra	495.303	0.124	**450.174**	**496.179**	4.520
Grafos	Neo4J	62.153	0.655	176.173	108.956	17.349

Tabla 1. Tabla de tiempos de carga y consulta de datos obtenidos
Fuente: Elaboración propia (2014).

Considerando los resultados obtenidos y las características propias de cada base de datos, se encuentra una relación entre los tiempos y el tipo de entidad de la que se tratan.

Aquellas entidades con valores estáticos, como son los productos y las personas, se comportaron bien al ser insertados en la base de datos tanto relacional (SQL, clásicas) como documental. Se puede ver, además, como la base de datos orientada a documentos tuvo mayor facilidad para incorporar las entidades que relacionan otros datos, como la relación de "amigos" entre personas y la de "compras", entre personas y productos. Esto hace valer el poder de la misma para generar relaciones entre los documentos.

En general, para el proceso de carga masiva, las bases de datos NoSQL parecen no ser adecuadas. Sin embargo, a la hora de realizar consultas -punto clave de un motor de procesamiento de datos-, se pueden observar buenos resultados, especialmente en la base de datos clave-valor, la cual está diseñada específicamente para realizar grandes consultas en poco tiempo.

Conclusiones

Se perciben como interesantes los resultados obtenidos, los cuales mostraron que para la cantidad y las estructuras de datos utilizados no existen diferencias significativas entre los tiempos resultantes en las bases de datos SQL y NoSQL.

Se puede estimar que aquellas bases de datos con esquemas más rígidos son adecuadas para manejar estructuras de datos con información descriptiva, sin demasiadas relaciones entre ellas. No obstante, dada la cantidad de datos utilizada, ésta resulta insuficiente para realizar una afirmación al respecto.

Considerando las limitaciones de las bases de datos relacionales, el movimiento NoSQL que está surgiendo parece adecuado para resolver una gran cantidad de problemas adjudicados al fenómeno de Big Data, resultando tipos de datos poco influyentes, en este caso, con la performance de los motores. Es necesario tener en cuenta estas características en relación con otras como la posibilidad de escalar almacenamiento y procesamiento, o la facilidad de modelado e interpretación, junto con los puntos débiles de estas tecnologías.

Todas estas consideraciones hacen intuitiva la idea de la implementación de modelos políglotas (Nayak et al., 2013) (Nance et al., 2013), fusionando las distintas bases de datos tanto SQL como NoSQL, aprovechando así las mejores características de cada una de ellas. De esta forma, los datos que por lo general son inmutables podrían implementarse de manera documental, aprovechando la libertad de esquemas, o bien de manera relacional o mediante columnas, si se necesita mantener mayor consistencia; mientras que para consultas inmediatas o recurrentes, se utilizarían bases de datos de clave-valor.

Como futura línea de trabajo se propone la incorporación de mayor volumen de datos, al igual que modelos políglotas en las pruebas y comparación de resultados.

Referencias

Sadalage, P. & Fowler, M. (2013) NoSQL Distilled, *A Brief Guide to the Emerging World of Polyglote*. Persistence Boston: Addison Wesley.

Hecht, R. (2011). *NoSQL Evaluation*. Recuperado de http://rogerking.me/wp-content/uploads/2012/03/DatabaseSystemsPaper.pdf

López, D. (2012). *Análisis de las posibilidades de uso de Big Data en las organizaciones, Universidad de Cantabria, Santander, España*. Recuperado de http://prezi.com/2rflalqhfntn/ análisisde-las-posibilidades-de-uso-de-big-data/

Ramírez, H. & Herrera, J. F. (2013). *Un viaje a través de bases de datos espaciales*. Recuperado de http://ingenieria.udistrital.edu.co/digital/index.php/redesdeingenieria/article/view/226/412

Nayak, A.; Poriya, A. & Poojary, D. (2013). Types of NOSQL Databases and its Comparison with Relational Databases. *International Journal of Applied Information Systems* (IJAIS), 5(4).

Strauch, C. & Kriha, W. (2011). *NoSQL databases*. Recuperado de http://coitweb.uncc.edu/~xwu/5160/nosqldbs.pdf

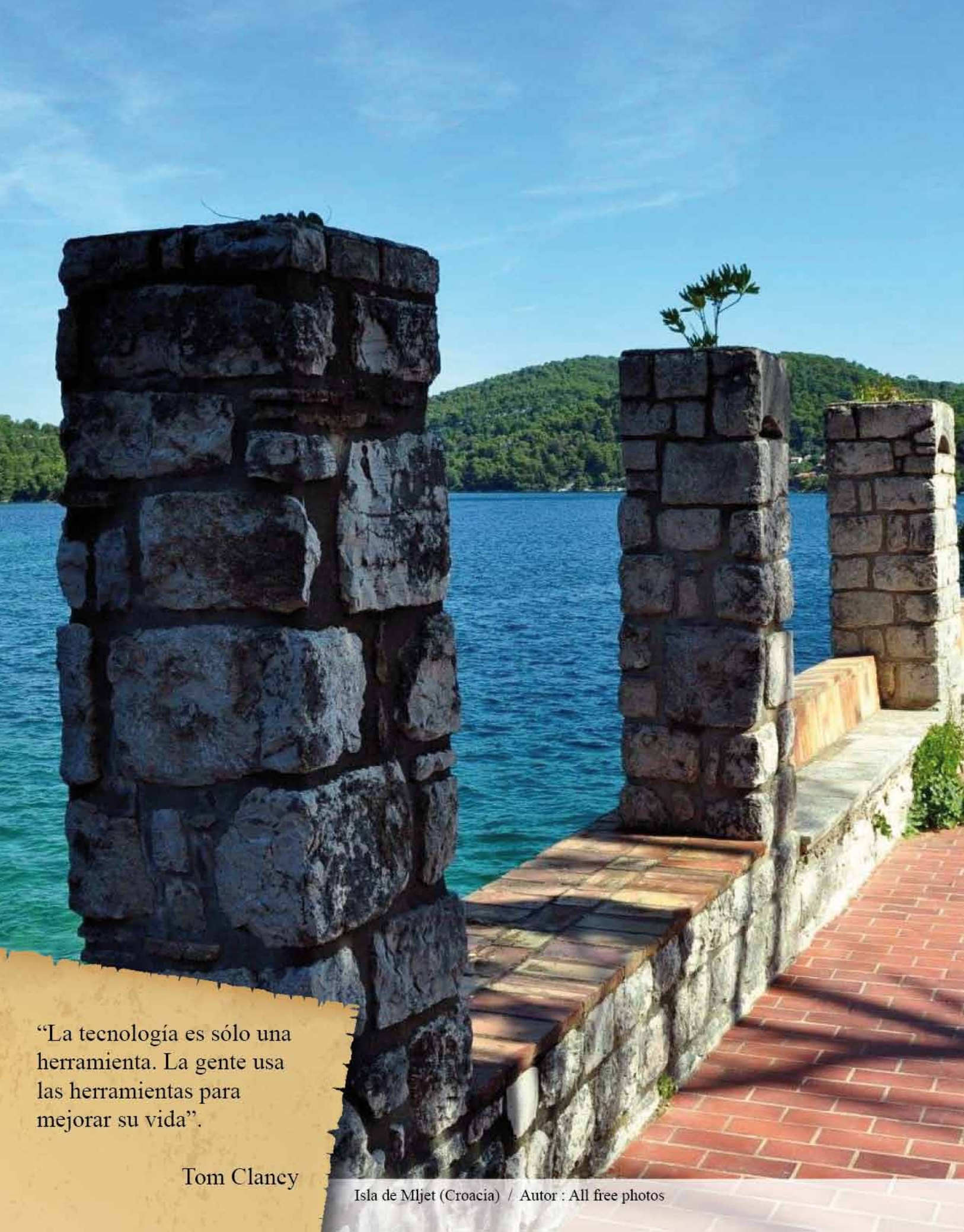
Del Busto, H. G. & Enríquez, O. Y. (2013). Bases de datos NoSQL. *Revista Telemática*, 11(3).

Bugiotti, F. & Cabibbo, L. (2013). *A Comparison of Data Models and APIs of NoSQL Datastores*. Recuperado de http://www.bugiotti.it/downloads/publications/noamSEBD13.pdf

Nance, C.; Losser, T.; Iype, R. & Harmon, G. (2013). *NoSQL vs RDBMS - Why There is Room for Both*. Recuperado de http://sais.aisnet.org/2013/Nance.pdf

Róttoli, G. D. (2014a). *Datos para comparativa de carga y consulta en bases de datos NoSQL*. Recuperado de http://tinyurl.com/lyy5thw

Róttoli, G.D. (2014b). *Resumen de códigos utilizados*. Recuperado de http://tinyurl.com/opwe425



"La tecnología es sólo una herramienta. La gente usa las herramientas para mejorar su vida".

Tom Clancy

Isla de Mljet (Croacia) / Autor : All free photos