

Revista Científica y Tecnológica UPSE

Rendimiento de MariaDB y PostgreSQL

Performance MariaDB and PostgreSQL



Anabel Pilicita Garrido ¹ <https://orcid.org/0000-0002-0796-7797>; Yolanda Borja López ¹ <https://orcid.org/0000-0001-5349-8461>; Gonzalo Gutiérrez Constante ¹ <https://orcid.org/0000-0003-1484-8558>

¹ Universidad Central del Ecuador, Ecuador

Resumen

El almacenamiento de información en las bases de datos es un tema importante para toda organización en la actualidad. Son los Sistemas Gestores de Bases de Datos conocidos como SGDB los encargados de ocultar la complejidad que conlleva la administración de la base de datos. Es común que cada día sean almacenados millones y millones de datos más aún porque en la actualidad existen nuevas tendencias y exigencias para el procesamiento de grandes volúmenes de datos a gran velocidad. Por lo tanto, es indispensable que los SGBD tengan un correcto rendimiento para hacer frente a estas nuevas exigencias. El presente trabajo muestra los resultados obtenidos al realizar pruebas de rendimiento en los Sistemas Gestores de Base de Datos MariaDB y PostgreSQL, aplicando la arquitectura de tres capas para los escenarios de prueba. Ambos SGBD son potentes y muy utilizados por diferentes sistemas. Los resultados obtenidos están relacionados con la cantidad de peticiones por segundo procesadas por estos sistemas. En este caso MariaDB fue superior que PostgreSQL a medida que aumentó la cantidad de usuarios que realizaban en diferentes escenarios gracias a sus motores de búsqueda que generan mejor velocidad para el procesamiento de datos.

Abstract

The storage of information in databases is an important issue for every organization today. They are the Database Management Systems known as SGDB in charge of hiding the complexity that the administration of the databases entails. It is common that millions and millions of data are stored every day even more because today there are new trends and demands for the processing of large volumes of data at high speed. Therefore, it is essential that the DBMS have a correct performance to face these new demands. This work shows the results obtained when performing performance tests on the MariaDB and PostgreSQL Database Management Systems, applying the three-layer architecture for the test scenarios. Both DBMS are powerful and widely used by different systems. The results obtained are related to the number of requests per second processed by these systems. In this case, MariaDB was superior to PostgreSQL as the number of users that performed in different scenarios increased thanks to its search engines that generate better data processing speeds.

Palabras clave:

Base de datos;
SGBD; SQL,
rendimiento

Keywords:

Database; SQL;
DBMS,
performance

Recibido: octubre/10/2020 **Aceptado:** noviembre/05/2020 **Publicado:** diciembre/21/2020

Forma de citar: Pilicita Garrido, A.; Borja López, Y.; Gutiérrez Constante, G. (2020). Rendimiento de MariaDB y PostgreSQL. Revista Científica y Tecnológica UPSE, 7(2) pág. 09-16. DOI: 10.26423/rctu.v7i2.538.

* Autor para correspondencia: anabelpilicita@gmail.com

1. Introducción

Hoy en día empresas, entidades u organizaciones de cualquier índole necesariamente administran grandes volúmenes de información de personas vinculadas a las mismas e información relevante tanto para los usuarios como para las mismas entidades. Dicha información debe ser debidamente almacenada y administrada por algún sistema para su posterior uso en distintas aplicaciones. Por tal motivo, surgen los Sistemas Gestores de Bases de Datos (SGBD) o Data Base Management System (DBMS) como un componente vital de todo sistema en la actualidad.

Los Sistemas Gestores de Bases de Datos, como su nombre lo indica son sistemas que generan independencia de datos, gestionan la concurrencia de acceso, garantizan la integridad de datos, proveen la persistencia de los mismos, enfrentan la recuperación frente a fallos y brindan la seguridad necesaria a toda la información que se encuentra almacenada en las bases de datos (Gunjal 2003). Las funciones de los SGBD son realmente importantes y su propósito general es ocultar toda la complejidad que conlleva el uso bases de datos a gran escala mejorando su administración.

Actualmente, son conocidos de forma general dos tipos de SGBD según el lenguaje que utilizan para definir los datos. Por una parte, los sistemas que utilizan el lenguaje de consulta estructurada o Structured Query Language (SQL); que en la vida cotidiana son conocidos como SGBD SQL o SGBD relacionales entre algunos ejemplos se encuentra: Oracle, MariaDB, PostgreSQL, Microsoft SQL Server entre otras.

Por otra parte, los sistemas que no usan el lenguaje SQL son conocidos como SGBD NoSQL los mismos que tienen una estructura dinámica para el almacenamiento de los datos entre algunos ejemplos están: MongoDB, Cassandra, Redis, entre otras (Acens 2014). Sistemas aplicados comúnmente para Big Data o Machine Learning nuevas tendencias que están marcando nuestro presente.

Son estas nuevas tendencias tecnológicas y el crecimiento de las actividades productivas las que han aumentado las necesidades de cómputo sin dejar de lado el rendimiento de todos los sistemas implementados. Por lo tanto, los SGBD enfrentan nuevos retos uno de ellos es la estructuración y gestión de grandes volúmenes de datos con el fin de tener una capacidad de almacenamiento óptima. El acceso concurrente de usuarios mediante consultas de forma que, si existe un crecimiento exponencial de usuarios, los SGBD deben ser capaces de lidiar con este tipo de situaciones manteniendo un adecuado rendimiento. Es una problemática que debe ser tratada y analizada para evaluar el desenvolvimiento de los SGBD en diferentes situaciones ya que en el mercado existe una gran variedad de estos sistemas y es necesario saber elegir uno de ellos de acuerdo a las necesidades de cada entidad.

Para el presente trabajo se toma en consideración a los SGBD SQL porque son sistemas tradicionales, pero en la actualidad han tenido que plantear nuevas soluciones frente al tratamiento de datos a gran escala debido a la competencia que se tiene frente con SGBD NoSQL que han abarcado y han tenido gran aceptación por su capacidad de procesar grandes volúmenes de datos. Se realizaron pruebas de rendimiento en dos SGBD que son MariaDB y PostgreSQL muy utilizados y populares en el mercado.

MariaDB está desarrollada por la fundación con el mismo nombre y es considerada una versión mejorada de MySQL, tiene alta compatibilidad con su antecesora y su mejora radica en ofrecer una mejor seguridad al momento de su implementación (Foundation 2019). Pero una de sus características más destacables es la incorporación de nuevos motores de búsqueda para proporcionar mayor escalabilidad y mejores velocidades al momento de realizar consultas a la base de datos.

Por otra parte, PostgreSQL está muy bien catalogado por su estabilidad, potencia, robustez y la facilidad de administración e implementación. Adicionalmente, utiliza un sistema cliente servidor con el uso de hilos para un procesamiento correcto de las consultas hacia la base de datos (Zea Ordóñez, 2017).

MariaDB como PostgreSQL son sistemas asequibles en nuestros días, los mismos que en el ranking mundial están entre los mejores puntuados teniendo el segundo y quinto lugar, respectivamente (DB-Engines Ranking 2020). Ambos son sistemas de código abierto por ello el énfasis en que son sistemas asequibles y con buenas características que evidencian su potencia y alto grado de rendimiento. El ranking también muestra que son usadas por una gran cantidad de administradores que buscan un SGBD para el funcionamiento integral de las aplicaciones que usan para almacenar los datos basados en el lenguaje SQL y de tipo relacionales.

El propósito del presente trabajo consiste en evidenciar el rendimiento tanto de MariaDB como de PostgreSQL bajo los mismos escenarios, mediante almacenamiento masivo de datos y realizando consultas a cada uno de los SGBD, registrando los tiempos que tarda cada uno por separado para comparar los resultados y determinar su rendimiento en las pruebas realizadas.

El documento se encuentra estructurado en la primera parte por los trabajos relacionados, bases de datos relacionales y el rendimiento que puede ser evidenciado por medio de diferentes consultas a los gestores de bases de datos. En la segunda parte se especifica la metodología y el escenario planteado para el desarrollo de las pruebas. En la tercera parte, se muestra y se realiza un análisis de los resultados obtenidos. Finalmente, se presenta las conclusiones del trabajo.

Según la investigación de Durán, Tandazo, Morales, Morales Cardoso que evaluó el rendimiento de bases de datos columnares, determinó que tanto MySQL como

PostgreSQL mantienen la relación entre el volumen de datos y el tiempo de respuesta es decir que se incrementa proporcionalmente, al contrario, en las bases de datos de la familia columnar Cassandra y MonetDB, los tiempos de ejecución no sufren variaciones notables entre el volumen de los datos y el tiempo (Durán 2019).

En la investigación de Portilla y Bernal donde se realizó la comparación del rendimiento de los gestores de base de datos Oracle y MySQL, se concluyó que quienes trabajan con gran cantidad de datos y gran cantidad de tablas es donde Oracle ofreció un mejor rendimiento. Se determinó que para procesos de agregar o eliminar datos la rapidez es superior de MySQL, mientras que Oracle es más rápido a la hora de consultar registros en bases de datos (Portilla Tovar and Bernal Gómez 2018).

Por otra parte, en el trabajo realizado por Torreno y Medrano, donde la comparativa se realizó con gestores como MySQL y Oracle determinaron que Oracle realiza consultas de forma más rápida, pero MySQL generó tiempos óptimos con actualización y eliminación de registros (Diez del Valle and Torreño 2016).

Finalmente, con respecto al análisis de Inzunza realizado con Sql Server 2012, PostgreSQL y MySQL para determinar su rendimiento se determinó que PostgreSQL tuvo mejores resultados de rendimiento en comparación a MySQL y Sql Server 2012.

1.1. Sistemas Gestores de Base de Datos Relacionales.

Un SGBD es una colección de programas que permiten a los usuarios crear y mantener una base de datos. En concreto un sistema de gestión de base de datos es un software que oculta la complejidad de gestión de los datos al desarrollador. Entre sus características principales tenemos (Gunjal 2003):

- Independencia: Los datos se gestionan independientemente de los programas que los vayan a utilizar teniendo así una independencia lógica y de los ficheros en los que se almacenen obteniendo una independencia física.
- Gestión de concurrencia de acceso.
- Integridad de datos: Mediante transacciones ACID y BASE.
- Persistencia.
- Recuperación frente a fallos
- Seguridad
- Rendimiento
- Lenguajes de Definición y Manipulación de Datos.

Los gestores de bases de datos relacionales son capaces de almacenar una gran cantidad información de naturaleza heterogénea y que guarda relaciones que pueden ser representadas formalmente según el modelo relacional (Martínez López & Gallegos Ruiz, 2017).

Las bases de datos relacionales han evolucionado de forma que hoy en día se puede hablar de los siguientes ejemplos: MariaDB, PostgreSQL, Oracle, Access, MySQL, SQL Server entre otras. Algunas de las mencionadas son propietarias y otras de código abierto (Amazon 2019).

Dentro de cada SGBD se definen las bases de datos donde se especifica los tipos de datos, la estructura y restricciones para los datos que se van a almacenar en la misma. La manipulación de la base de datos incluye funciones tales como consultar la base de datos para recuperar unos datos específicos, actualizar base de datos entre otras operaciones que se lo realiza mediante el lenguaje SQL (Pilicita 2016).

Como se mencionó anteriormente, para realizar operaciones sobre las bases de datos, ya sea, definirla, crear tablas, realizar modificaciones (Martín Escofet 2013) es usado el lenguaje SQL que permite encontrar la información exacta de una manera rápida, repetible y confiable. De esta manera es posible consultar fácilmente y extraer datos significativos a partir de grandes bases de datos (Pilicita 2016).

1.2. Rendimiento

Son las aplicaciones ya sean de escritorio, web o móviles que disponen de una interfaz gráfica para la interacción del usuario final con los sistemas complejos y son transparentes para la gran mayoría de personas que utilizamos servicios. Es el usuario quien siempre espera un acceso rápido y oportuno a la información requerida que en la gran mayoría de los casos se encuentra almacenada en bases de datos. Sin duda el usuario final es exigente y no le gusta esperar por lo que los tiempos de respuesta de los SGBD deben ser acorde a las necesidades actuales.

Según el estudio realizado por la Dynatrace en donde participaron distintos países como Estados Unidos, Alemania, Australia y Brazil. Se determinó que: “Una sola transacción web implica la participación de un promedio de 35 sistemas o componentes de tecnología diferentes” (Dynatrace 2018). Dentro del promedio de 35 sistemas en este ejemplo sin duda alguna está presente los SGBD los cuales se encuentran conectadas para realizar consultas y extraer información almacenadas en las mismas y mostrar los datos de forma visual; o almacenar información dentro de las bases de datos.

Estos procesos acarrearán determinados tiempos bajo diferentes escenarios lo que influye en el rendimiento global ya que si los tiempos son altos de procesamiento en una base de datos su resultado está reflejado en tiempos de espera mayores en las aplicaciones lo que significa que el usuario espera más tiempo y su experiencia no va a ser la más grata.

En las pruebas de rendimiento se consideran tres características como es el comportamiento de en el

tiempo, uso de recursos y la capacidad de peticiones (Maila 2017). A continuación, se presenta el enfoque de una prueba de carga las mismas que permiten identificar los problemas que afectan el rendimiento.

- La prueba de rendimiento: Identifica la velocidad del sistema cuando se realiza una carga. Para lo cual los tiempos de respuesta son importantes en cada transacción que realice un usuario (Molyneaux 2009).
- Prueba de estrés: Esta prueba se utiliza normalmente para romper la aplicación ya sea doblando el número de usuarios para que sean agregados a la aplicación y se ejecute una prueba de carga hasta que se rompe (Falc 2011).

Es importante recalcar que no solo para los usuarios finales representa un inconveniente los tiempos de respuesta, sino también para las empresas que tienen que asumir con los costos que implica la pérdida de productividad por no disponer del rendimiento adecuado. De tal forma que se debe actuar tanto en la infraestructura en los componentes que conforman su sistema y uno de los más importantes es la selección de los SGBD según su aplicación (Performer 2017).

2. Materiales y Métodos

El plan de pruebas en la presente investigación tiene como propósito medir tiempos de respuesta frente a diferentes pruebas de carga en los distintos escenarios, para su posterior análisis e ilustración mediante gráficos y tablas. La metodología utilizada fue en base a la arquitectura en tres capas que consta de los siguientes componentes (Acosta Gonzaga, Álvarez Cedillo, and Gordillo Mejía 2006).

- Presentación: Es la interfaz presentada al usuario para lo cual se creó una serie de páginas web presentada al usuario final la misma que permite realizar consultas a las bases de datos del lado del usuario y visualizar resultados
- Reglas de Negocio: Capa donde se maneja la lógica para enlazar a la capa de presentación con la de almacenamiento. Por lo tanto, para las pruebas se utilizó Apache para el alojamiento de la página web que accede el usuario final y a su vez están conectadas hacia bases de datos creadas en MariaDB y PostgreSQL.
- Almacenamiento: Encargada de la persistencia de los datos, suministra y almacena la información. En este caso se creó la misma estructura de datos en ambos SGBD.

Luego de tener como referencia a la arquitectura en tres capas, se tiene claro que se debe estructurar la misma con herramientas que permitan su desarrollo. Para las pruebas de rendimiento, es difícil de conseguir que las condiciones de prueba sean similares a las esperadas en el uso real. Debido a que los sistemas en producción

tienen un carácter aleatorio de la carga de trabajo y aunque en las pruebas se intente dar lo mejor de sí para imitar el volumen de trabajo que pueda tener el entorno de producción, es imposible reproducir exactamente la variabilidad en condiciones reales.

Con este antecedente, se establece que el equipo donde se realizaron las pruebas consta de las siguientes características técnicas:

- Portátil HP Pavilion dm4 con
- Procesador Corei5.
- Memoria RAM de 8GB.
- Sistema Operativo Windows 10 de 64bits

Y las versiones del software fueron las siguientes:

- Servidor Web Apache 2.4.17
- Servidor MariaDB 10.1.10
- Servidor PostgreSQL 9.5
- Apache Jmeter versión 2.13

Cabe recalcar que Jmeter el software utilizado para realizar pruebas de concurrencia y estrés, permite simular el acceso de usuarios, por lo que se logra tener un entorno con múltiples usuarios realizando peticiones hacia los sistemas gestores de base de datos, sin que esto implique que sean usuarios reales.

Se desarrolló un sitio web con varias páginas web por medio del mapa web se muestra el esquema conceptual que muestra el acceso a cada página web. Como se puede ver en la Figura 1.

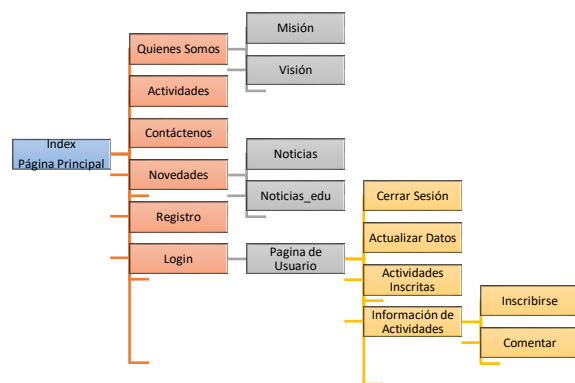


Figura 1. Mapa del Sitio Web.

Para seleccionar las páginas web se tomó en cuenta que sean páginas dinámicas que tengan incrustado código PHP para conexión a base de datos. Adicionalmente, se tomó en cuenta aquellas páginas que consten de formulario de registro para realizar la inserción de datos y las páginas que permitan realizar consultas, todo esto dirigido a una base de datos. Con base a estas consideraciones se seleccionó dos páginas web que

cuentan con código donde se realicen consultas hacia las bases de datos. Las páginas que fueron utilizadas para el plan de pruebas son actividades5.php por medio de la cual se realizan una consulta y contacto.php que inserta datos en las bases de datos creadas en ambos SGBD.

Con respecto al diseño de la estructura de las bases de datos se partió del diseño en base al modelo conceptual, en este caso el modelo Entidad-Relación como se ha mencionado anteriormente, tanto MariaDB como PostgreSQL son gestores relacionales que cuentan con entidades y sus respectivos atributos visualizados en la Figura 2.

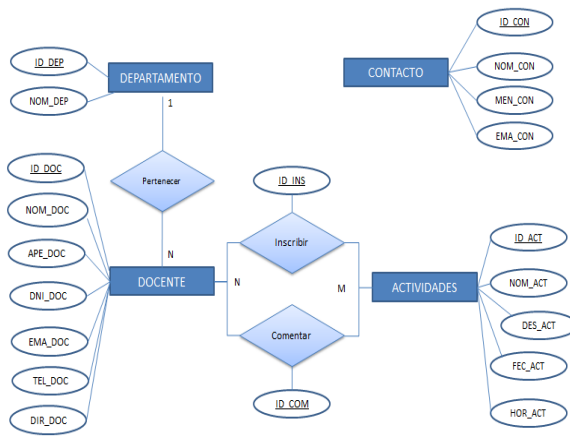


Figura 2. Modelo Entidad-Relación aplicado a los SGBD

Cabe recalcar que tanto MariaDB como PostgreSQL disponen del mismo esquema de base de datos creadas y son las variables de entorno en el esquema para la realización de las distintas pruebas.

Finalmente, se utilizó Apache Jmeter versión 2.13 software la que permite realizar pruebas de rendimiento y fue desarrollado en Java como una herramienta totalmente de código abierto. Es compatible para entornos tanto estáticos, dinámicos y sobre todo en páginas web y permite evaluar el rendimiento aplicando pruebas de carga y estrés. (Apache Foundation Software 2019).

La herramienta Apache Jmeter permitió realizar pruebas de consulta a la base de datos utilizando la página web actividades5.php y pruebas de carga de información con contacto.php. Conectadas a una base de datos en MariaDB, y luego se realizó las mismas pruebas en otra base de datos en PostgreSQL.

Para lograr visualizar cambios significativos en los tiempos de respuesta en ambos sistemas gestores de base de datos, se debe ir incrementando la cantidad de usuarios que acceden a las mismas, esto debido a que un sistema debe ser capaz de escalar con el aumento de peticiones. Por lo tanto, los escenarios están constituidos de menor a mayor desde 10, 50, 100 y 150

usuarios para demostrar la concurrencia de usuarios a medida que aumenta la cantidad de peticiones.

Las pruebas de carga fueron realizadas mediante peticiones hacia las páginas web con el objetivo de visualizar el rendimiento a medida que aumenta el número de usuarios (Pilicita 2016). Primero para un acceso simultáneo por parte de los usuarios visualizando información almacenada en las bases de datos y luego, para transacciones durante el tiempo que dura la carga de la página.

3. Discusión y Resultados

En este apartado se muestran los resultados obtenidos luego de realizar las diferentes pruebas para evidenciar el rendimiento tanto de PostgreSQL y MariaDB en distintos escenarios. En este caso son 10 los usuarios que realizan una consulta hacia los SGBD al mismo tiempo y como se observa en la Figura 3, donde las peticiones realizadas han sido atendidas en su totalidad y el rendimiento en ambos es igual por lo tanto no existe una diferencia en la cantidad de peticiones atendidas.

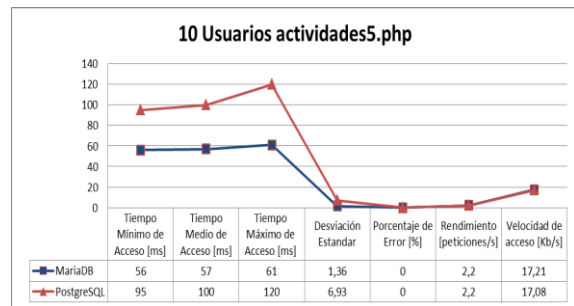


Figura 3. Gráfico 10 usuarios actividades5.php PostgreSQL y MariaDB

En el caso de 50 accesos concurrente se visualiza en la Figura 4, para realizar consultas simultáneamente, se mantiene el número de peticiones procesadas que es de 10 peticiones por segundo, por lo tanto, el rendimiento se mantiene en ambos casos con un porcentaje de error del 0%. Si bien es cierto en este caso como el anterior los tiempos varían, pero no lo suficiente para indicar que el rendimiento es superior.

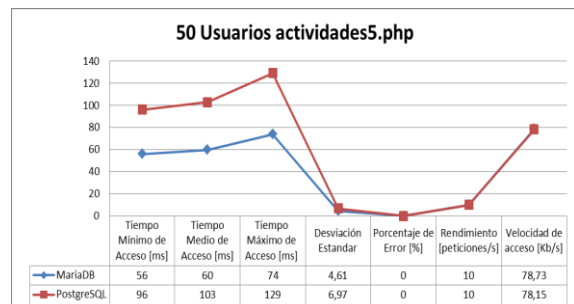


Figura 4. Gráfico 50 usuarios actividades5.php PostgreSQL y MariaDB

En la Figura 5 con 100 usuarios conectados simultáneamente los tiempos empiezan a variar significativamente por lo que en número de peticiones que procesa MariaDB es 19,8 peticiones por segundo y en el caso de PostgreSQL es de 19,7 peticiones por segundo es ligeramente superior MariaDB en rendimiento.

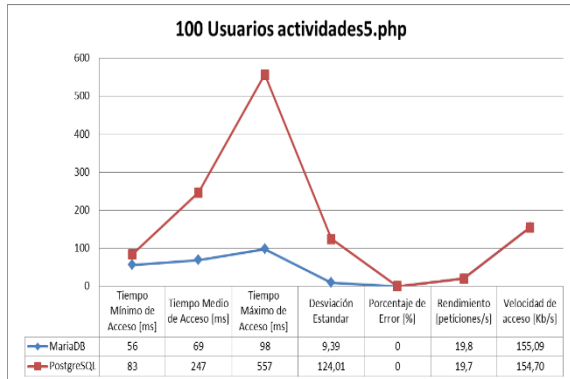


Figura 5. Gráfico 100 usuarios actividades5.php PostgreSQL y MariaDB.

Los resultados con 150 usuarios conectados y realizando consultas es evidente que los tiempos de acceso son menores en MariaDB lo que genera la posibilidad de procesar mayor cantidad de peticiones por segundo. No cabe duda que MariaDB procesa alrededor de 5 peticiones más por segundo que PostgreSQL visualizado en la Figura 6. Es decir a medida que aumentan los usuarios MariaDB es consistente y más robusta.

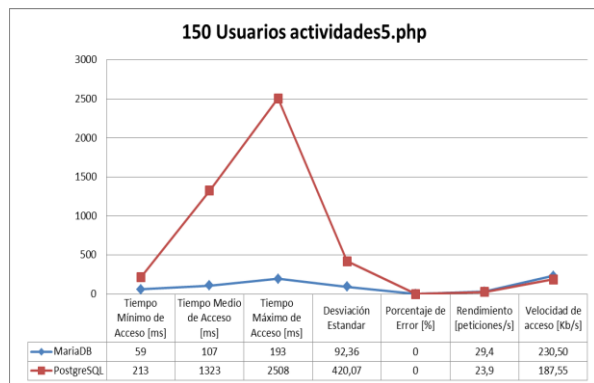


Figura 6. Gráfico 150 usuarios actividades5.php PostgreSQL y MariaDB.

Como se mencionó en el plan de pruebas se toma en consideración pruebas de carga donde por medio de un formulario establecido en la página web contacto.php se almacenan información en los SGBD. Por lo tanto, en la Figura 7 con 10 usuarios ingresando información al mismo tiempo el rendimiento se mantiene y se procesan las peticiones sin errores a una velocidad similar en ambos casos.

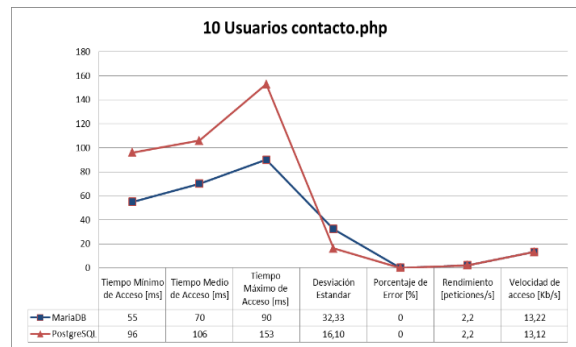


Figura 7. Gráfico 10 usuarios contacto.php PostgreSQL y MariaDB

Por otra parte, al aumentar el número de usuarios a 50 en la Figura 8 demuestra que cuando se trata de almacenar datos el rendimiento se mantiene y son procesadas 10 peticiones por segundo en los dos SGBD.

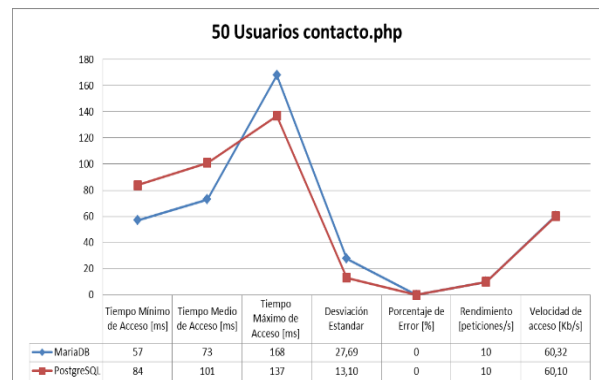


Figura 8. Gráfico 50 usuarios contacto.php PostgreSQL y MariaDB

En la Figura 9 cuando se dispone de 100 usuarios enviando información simultáneamente el rendimiento se mantiene existen cambios en los tiempos de respuesta, pero no lo suficientemente amplios para crear una brecha diferente.

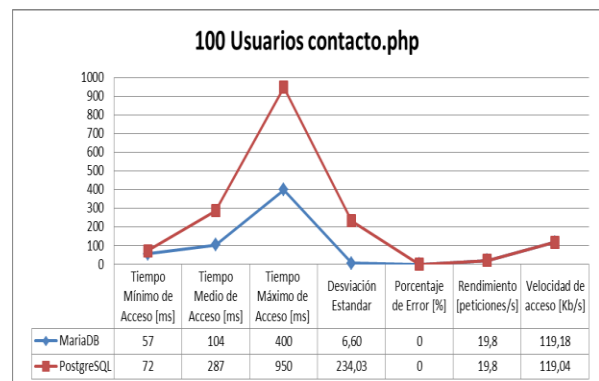


Figura 9. Gráfico 100 usuarios contacto.php PostgreSQL y MariaDB

Finalmente, con el escenario de 150 usuarios el

rendimiento es mejor con MariaDB consigue procesar una petición y media más que PostgreSQL, evidenciando una mayor velocidad en el procesamiento de peticiones.

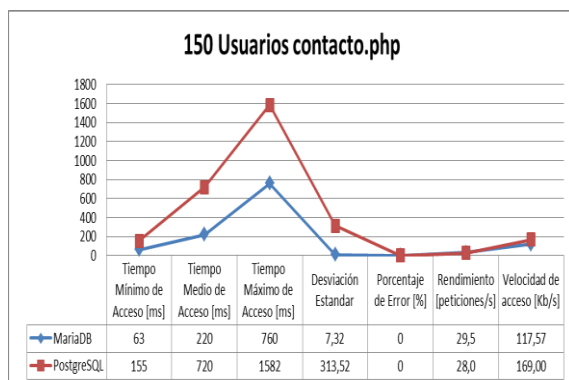


Figura 10. Gráfico 150 usuarios contacto.php PostgreSQL y MariaDB

Los resultados muestran que los tiempos de acceso aumentan a medida que incrementa el número de usuarios que realizan peticiones a los SGBD. Se dispone de un tiempo mínimo, promedio y máximo de acceso que están estrechamente relacionados con la velocidad de acceso. Por lo tanto, si la velocidad es óptima estos tiempos de espera para acceder son menores en todos los escenarios presentados.

Es importante recalcar que el porcentaje de error registrado en todas las pruebas fue del 0% lo que significa que todas las peticiones realizadas por los usuarios finales fueron exitosas. Es decir todas las consultas de búsqueda en las bases de datos fueron atendidas; así como el almacenamiento generado por los usuarios dentro de las bases de datos en ambos SGBD.

En el caso de consultas hacia las bases de datos con 10 y 50 usuarios las peticiones realizadas por los SGBD son iguales, aunque desde un inicio la velocidad de MariaDB es ligeramente más rápida que PostgreSQL. Los resultados cambian a mitad que aumenta el número de peticiones entre 100 y 150 usuarios la velocidad de procesamiento superior de MariaDB permite que la misma pueda procesar más peticiones mejorando su rendimiento en comparación con PostgreSQL.

En el caso de ingresar datos masivamente en los SGBD específicamente con 10, 50 y 100 usuarios el número de peticiones atendidas fueron iguales. Por lo tanto, el rendimiento se mantiene tanto con MariaDB como en PostgreSQL; pero los resultados varían cuando el número de usuarios asciende a 150 usuarios logrando procesar más peticiones por segundo MariaDB marcando una diferencia con PostgreSQL nuevamente por presentar mejores velocidades de acceso.

En los resultados obtenidos no cabe duda de que los dos SGBD testeados en estos escenarios son potentes y

robustos. Los tiempos de respuesta son bajos y tienen gran capacidad de adaptarse a diferentes escenarios. Pero no cabe duda que MariaDB es una versión mejorada de MySQL que ha incorporado nuevos motores de búsqueda para enfrentar nuevos retos en procesamiento de grandes volúmenes de datos. La velocidad de estos motores genera que a medida que aumentan los usuarios pueda responder de forma oportuna y mantener un rendimiento aceptable lo que le permite estar a la vanguardia y competir con otros SGBD en las pruebas realizadas se pone a consideración un mejor rendimiento por parte de MariaDB.

4. Conclusiones

En todos los escenarios donde se realizaron pruebas de consulta a la base de datos y pruebas de carga de información evidenciaron el comportamiento de dos Sistemas Gestores de Base de datos SQL bajo las mismas condiciones. Ambos sistemas tienen gran potencia de búsqueda y gran capacidad de escalabilidad ya que el porcentaje de errores fue de 0%. Por otra parte, los resultados obtenidos muestran que los tiempos de espera para acceder a las bases de datos aumentan a medida que la cantidad de usuarios se incrementa esto es evidente en los tiempos de acceso, entre menos cantidad de usuarios intentan acceder simultáneamente la respuesta es más rápida por parte de los SGBD.

En el caso de 10 usuarios que realizan consultas hacia los sistemas gestores de base de datos la velocidad de acceso fue de 17,21 Kb/s con MariaDB y 17,08 Kb/s con PostgreSQL esta diferencia no es significativa y se consigue procesar en ambos casos 2,2 peticiones por segundo. Con 50 usuarios la velocidad de acceso fue de 78,73 Kb/s con MariaDB y 78,15 Kb/s con PostgreSQL y en ambos casos realizaron 10 peticiones por segundo. Para 100 usuarios la velocidad de acceso fue de 155,09 Kb/s con un total de 19,8 peticiones por segundo en MariaDB y 154,70 Kb/s en la velocidad de acceso y 19,7 peticiones por segundo en PostgreSQL. Hasta este punto el rendimiento en ambos gestores es el mismo porque logran procesar la misma cantidad de peticiones sin ningún error. Pero a medida que se aumenta la cantidad de usuarios los valores cambian esto se evidenció al tener 150 usuarios conectados solicitando consultas de forma simultánea donde la velocidad de acceso en el caso de MariaDB fue de 230,50 Kb/s y logra procesar 29,4 peticiones por segundo y PostgreSQL tiene una velocidad de acceso de 187,55 Kb/s y procesa 23,9 peticiones por segundo. Por lo tanto, al realizar específicamente una transacción de consultas hacia una base de datos bajo el escenario de pruebas establecido para este trabajo se pudo constatar que a medida que aumenta la cantidad de usuarios MariaDB demuestra tener un mejor rendimiento en comparación con PostgreSQL por la cantidad de peticiones que puede procesar.

Por otra parte, en el escenario de 10 usuarios que

insertaron datos de forma simultánea en los sistemas gestores la velocidad de acceso para MariaDB fue de 13,22 Kb/s y para PostgreSQL fue de 13,12 Kb/s. La diferencia no es significativa, es por ello que se mantuvo el valor de 2,2 peticiones por segundo en ambos casos. Para 50 usuarios la velocidad de acceso fue de 60,32 Kb/s en MariaDB y 60,10 Kb/s en PostgreSQL, el rendimiento se mantuvo con 10 peticiones por segundo sin cambios. Para 100 usuarios tampoco existió cambios significativos con 19,8 peticiones por segundo para los dos sistemas con una velocidad en MariaDB de 119,18 Kb/s y en PostgreSQL de 119,04 Kb/s. Finalmente, al tener 150 usuarios se observó que cuando se almacenan datos en MariaDB atiende 29,5 peticiones por segundo y en el caso de PostgreSQL es capaz de procesar 28,5 peticiones por segundo. La diferencia es de una petición específicamente en este escenario de prueba. El margen no es amplio pero esto demuestra que mientras aumenta el número de usuarios y la diferencia tiende a aumentar, conforme sea mayor el número de usuarios que realizan alguna transacción hacia los gestores MariaDB atiende más peticiones por segundo.

Los SGBD SQL son conocidos como bases de datos tradicionales, pero actualmente son los SGBD NoSQL los que toman protagonismo por ser capaces de escalar sin ningún problema. Como línea futura se puede generar un escenario para evaluar el rendimiento entre sistemas SQL vs NoSQL para constatar los tiempos de respuestas en los mismos.

5. Referencias

- [1] Acens. 2014. "Bases de Datos NoSQL . Qué Son y Tipos Que Nos Podemos Encontrar." 7.
- [2] Acosta Gonzaga, Elizabeth, Jesús Antonio Álvarez Cedillo, and Abraham Gordillo Mejía. 2006. "Arquitecturas En N-Capas: Un Sistema Adaptivo." *Polibits* 34:34–37.
- [3] Amazon, AWS. 2019. "Bases de Datos SQL | AWS." Retrieved February 6, 2020 (<https://aws.amazon.com/es/relational-database/>).
- [4] Apache Foundation Software. 2019. "Apache JMeter - Apache JMeter™." Retrieved February 1, 2020 (<http://jmeter.apache.org/>).
- [5] DB-Engines Ranking. 2020. "Ranking of Database Management Systems." Retrieved April 3, 2020 (<https://db-engines.com/en/ranking>).
- [6] Díez del Valle, Alvaro, and Diego Torreño. 2016. "Comparativa de Rendimiento de Consultas Entre Sistemas Relacionales Oracle y MySQL."
- [7] Durán, Jhonatan, Eduardo Tandazo, Mario Morales, and Santiago Morales. 2019. "Ingenius." *Ingenius. Revista de Ciencia y Tecnología* (22):47–58.
- [8] Dynatrace, Enterprice. 2018. "Estudio de Páginas Web." Retrieved February 6, 2020 (<https://www.dynatrace.com/news/press-release/el-76-de-las-empresas-vaticina-que-en-poco-tiempo-sera-imposible-gestionar-el-rendimiento-de-las-aplicaciones-de-negocio-por-la-complejidad-tecnologica/>).
- [9] Falc, Diego Alberto. 2011. "Comparativa de Herramientas Para Aplicar Pruebas de Estrés y Desempeño Para Aplicaciones Web."
- [10] Foundation, Mariadb. 2019. "MariaDB." 1–5.
- [11] Gunjal, Bhojaraju. 2003. "Database Management: Concepts and Design." In *Proceedings of Knowledge Management in Special Libraries in Digital Environment : XXIV All India Conference of IASLIC, 15-18 December (February)*:385–98.
- [12] Maila, Fernando. 2017. "Evaluación de Herramientas Open Source Para Pruebas de Fiabilidad y Rendimiento de Aplicaciones Web."
- [13] Martín Escofet, Carmen. 2013. "El Lenguaje SQL." 1–42.
- [14] Molyneaux, Ian. 2009. *The Art of Application Performance Testing*. O'Reilly Media, Inc.
- [15] Performer, Silk. 2017. "Pruebas de Carga Resumen Ejecutivo."
- [16] Pilicita, Anabel. 2016. "Diseño e Implementación de Una Aplicación Web, Para Su Posterior Análisis Mediante Dos Tipos de Base de Datos." 1–100.
- [17] Portilla Tovar, Johan Alexander, and Mireya Bernal Gómez. 2018. "Comparación Del Rendimiento de Los Comandos Insert, Select y Delet En Los Sistemas Gestores de Bases de Datos Oracle y MYSQL." *Inventum* 12(23):10–21.
- [18] Zea Ordóñez, Mariuxi Paola, Jimmy Rolando Molina Ríos, and Fausto Fabían Redrován Castillo. 2017. *Administración De Bases De Datos Con Postgresql*.