

ANALISIS PERBANDINGAN PERFORMA DATABASE SQL DB ORACLE DENGAN NOSQL DB MONGODB

Imelda Audina

Universitas Pembangunan Nasional “Veteran” Jawa Timur

audinaimelda3@gmail.com

Abstract

Databases have become an essential component of data management systems. In the ever-evolving information age, information and technology professionals are faced with the question of choosing a database that suits their diverse business and application needs. DBMS is divided into two types, namely relational database (SQL DB) and non-relational database (NoSQL DB). This research uses Oracle as a relational database and MongoDB as a non-relational database. The mechanism of data processing and storage is the difference between SQL DB and NoSQL DB. To find out the differences, especially in the query response from both, testing is needed. The database uses the gravity_book database from the cust_order table with a total data of 7550 rows and consists of 5 columns. Testing is done by applying four DML command operations in order to access and manipulate data from the cust_order table, namely by SELECT (reading data), UPDATE (updating data), INSERT (adding data), DELETE (deleting data) by limiting the number of different data records (750, 1500, 2550, 3000). This research aims to analyze the performance comparison between SQL DB Oracle and NoSQL DB MongoDB from the resulting query speed. The results show that the fastest response time for all test query commands is Oracle, while MongoDB ranks second after Oracle.

Keywords: Performance, SQLDB, NoSQLDB, Oracle, MongoDB.

Abstrak

Database telah menjadi komponen penting dalam sistem pengelolaan data. Di era informasi yang terus berkembang, para profesional di bidang informasi dan teknologi akan dihadapkan dengan pertanyaan mengenai pemilihan database yang sesuai dengan kebutuhan bisnis dan aplikasi yang sangat beragam. DBMS terbagi menjadi dua tipe, yakni basis data relasional (SQL DB) dan basis data non-relasional (NoSQL DB). Penelitian ini menggunakan Oracle sebagai basis data relasional dan MongoDB sebagai basis data non-relasional. Mekanisme pengolahan maupun penyimpanan data menjadi perbedaan antara SQL DB dengan NoSQL DB. Untuk mengetahui perbedaan khususnya pada respon kueri dari keduanya, perlu adanya pengujian. Database yang digunakan menggunakan database gravity_book dari tabel cust_order dengan jumlah data sebanyak 7550 rows dan terdiri dari 5 kolom. Pengujian yang dilakukan dengan menerapkan empat Perintah perintah DML agar dapat mengakses maupun memanipulasi data dari tabel cust_order yaitu dengan SELECT (membaca data), UPDATE (memperbarui data), INSERT (menambah data), DELETE (menghapus data) dengan membatasi jumlah record data yang berbeda-beda (750, 1500, 2550, 3000). Penelitian ini bertujuan untuk melakukan analisis perbandingan performa antara

SQL DB Oracle dan NoSQL DB MongoDB dari kecepatan *query* yang dihasilkan. Hasil penelitian menunjukkan bahwa *respon time* tercepat untuk seluruh perintah pengujian kueri adalah Oracle, sedangkan MongoDB menempati posisi urutan kedua setelah Oracle.

Kata Kunci : Performa, SQLDB, NoSQLDB, Oracle, MongoDB

PENDAHULUAN

Dalam era informasi yang terus berkembang, *database* menjadi komponen yang krusial dalam sistem pengelolaan data. Para profesional di bidang teknologi informasi dan pengembang aplikasi sering dihadapkan pada pertanyaan tentang pemilihan database yang sesuai untuk memenuhi kebutuhan bisnis dan aplikasi yang beragam. Sistem manajemen basis data (DBMS) dapat mengelola basis data secara efektif, yang merupakan sebuah sistem yang digunakan untuk mengelola kumpulan data terintegrasi dari berbagai entitas yang terhubung (Selviana & Roji, 2023). DBMS terbagi menjadi dua tipe, yakni basis data relasional (SQL DB) dan basis data non-relasional (NoSQL DB) (Fadli, Zulfa, Nugraha, Taryana, & Aliim, 2020). Mekanisme pengolahan maupun penyimpanan data menjadi perbedaan antara SQL DB dengan NoSQL DB atau Not Only SQL DB (Zhang, Chen, Ooi, Tan, & Zhang, 2015).

SQL DB menerapkan prinsip ACID (*Atomic, Consistent, Isolation and Durability*) sebagai proses penyimpanan maupun pengolahan data (Fadli, Zulfa, Nugraha, Taryana, & Aliim, 2020). *Atomic* merupakan prinsip yang jika terjadi kegagalan dalam transaksi, sistem harus memastikan bahwa semua Perintah dalam transaksi dibatalkan (*rollback*), sehingga basis data tetap konsisten dengan kata lain dapat menyebabkan kegagalan keseluruhan transaksi. Prinsip *consistency* menjamin bahwa setiap transaksi harus memenuhi semua aturan dan batasan yang telah ditetapkan. Jika transaksi melanggar aturan, maka transaksi tersebut akan dianggap gagal dan perubahan akan dibatalkan (*rollback*). Prinsip *isolation* menjamin bahwa ketika beberapa transaksi berjalan secara bersamaan, hasil akhirnya sama dengan jika transaksi tersebut dijalankan secara berurutan. Dengan kata lain, transaksi tidak akan terpengaruh oleh transaksi lain yang sedang berlangsung. *Durability* menjamin bahwa setiap perubahan yang dilakukan oleh transaksi akan tetap ada dan aman meskipun terjadi kegagalan sistem. Setelah transaksi selesai (*commit*), perubahan akan disimpan secara permanen di dalam basis data dan dapat dipulihkan jika diperlukan (Maukar, Marisa, Vitianingsih, & Puspitarini, 2022). Data yang tersimpan pada basis data relasional berbentuk tabel yang terdiri dari baris dan kolom (Fadli, Zulfa, Nugraha, Taryana, & Aliim, 2020). Adapun karakteristik SQL DB, antara lain data diorganisir dalam tabel relasional dengan skema yang ditentukan sebelumnya, struktur data yang terstruktur dan memiliki hubungan yang didefinisikan dengan jelas, memiliki konsistensi dan integritas data yang tinggi, cocok untuk aplikasi yang membutuhkan transaksi yang kompleks dan relasi antara entitas data (Hartanto,

Ardiansyah, & Kurniasih, 2024). Contoh SQL DB seperti MySQL, Oracle Database, DB2, Sybase, Microsoft SQL Server dan MS Access.

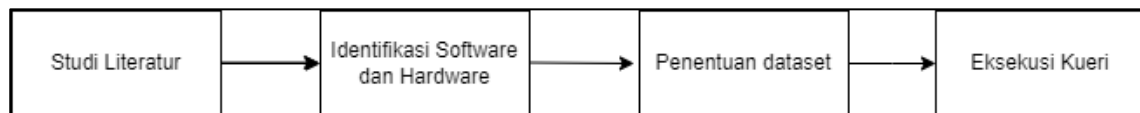
NoSQL DB menerapkan prinsip BASE, yang merupakan singkatan dari *Basically Available, Soft State and Eventual Consistency*. *Basically Available* menekankan bahwa sistem harus tetap tersedia dan responsif bahkan jika hasil yang diberikan tidak sepenuhnya konsisten atau lengkap. *Soft State* merupakan bahwa status sistem dapat berubah seiring waktu atau tidak konsisten sementara meskipun sistem sedang dalam proses pembaruan atau replikasi. *Eventual Consistency* menyatakan bahwa pada akhirnya sistem akan mencapai konsistensi data (Maukar, Marisa, Vitianingsih, & Puspitarini, 2022). Dalam basis data NoSQL, replikasi data dan propagasi perubahan membutuhkan waktu tertentu, dan dalam beberapa situasi, konsistensi data secara langsung tidak diperlukan. Oleh karena itu, basis data NoSQL lebih mementingkan ketersediaan dan skalabilitas daripada konsistensi yang segera dan kuat. Basis data NoSQL juga dikenal sebagai basis data awan atau big data yang dirancang untuk mengatasi masalah performa basis data relasional dalam memproses data tidak terstruktur maupun semi-terstruktur, seperti dokumen, email, multimedia, atau media social (Okman, Gal-Oz, Gonen, Gudes, & Abramov, 2011). Basis data ini menerapkan beragam teknik baru yang meliputi key-value, orientasi dokumen, grafik, dan sejenisnya (Tavares, Inacio, Rangkoly, & Sarah Bunda Desy Bawan, 2020). Contoh NoSQL DB seperti MongoDB, Cassandra, dan Redis.

Performa yang baik pada *database* ditandai dengan *response time* yang singkat, pemrosesan transaksi yang cepat akibat dari kapasitas perangkat lunak dan perangkat keras yang digunakan, dan eksekusi perintah yang diberikan. Beberapa faktor yang mempengaruhi performa database meliputi, waktu awal perintah ke sistem hingga respons muncul dengan kata lain terkait kecepatan waktu respon, kemampuan keseluruhan komputer untuk memproses data, konfigurasi hardware yang tepat dan penggunaan perangkat keras yang berkualitas, dan total memori yang diperlukan untuk menjalankan eksekusi dimana nilai ini diambil setelah eksekusi selesai (Hartono & Masyhur, 2023).

Dalam penelitian ini menggunakan Oracle sebagai basis data relasional dan MongoDB sebagai basis data non-relasional. Berdasarkan survei terbaru pada situs db.engines, diperoleh data SQL DB yang paling banyak digunakan adalah Oracle sedangkan NoSQL DB pada posisi pertama ditempati oleh MongoDB (DB-Engines Ranking, 2025). Survei tersebut mendukung penelitian ini dengan menggunakan Oracle dan MongoDB sebagai contoh untuk masing-masing tipe DBMS. Tujuan penelitian ini dibuat untuk melakukan analisis perbandingan performa antara SQL DB Oracle dan NoSQL DB MongoDB. Selain itu, memberikan pemahaman yang mendalam tentang perbedaan performa seperti kekuatan dan kelemahan masing-masing jenis database yang dapat membantu pengembang aplikasi dan pengolah data dalam membuat keputusan yang tepat dalam memilih database yang sesuai dengan kebutuhan.

METODE PENELITIAN

Pada dasarnya *database* relasional seperti SQL DB Oracle memiliki banyak perbedaan dengan *database* non relasional seperti NoSQL MongoDB. Untuk mengetahui perbedaan khususnya pada *response time* kueri dari keduanya, perlu adanya pengujian. Pengujian yang dilakukan dengan mengeksekusi empat perintah-perintah DML agar dapat mengakses maupun memanipulasi data yaitu dengan *select* (membaca data), *update* (memperbarui data), *insert* (menambah data), *delete* (menghapus data). Pada penelitian ini terdapat beberapa pendekatan dengan tahapan sebagai berikut :



Gambar 1. Tahapan Penelitian

Pada penelitian ini melakukan studi literatur untuk mengumpulkan dan menganalisis informasi mengenai basis data relasional dan non relasional seperti Oracle dan MongoDB. Pada tahap ini merupakan alat untuk pembelajaran dan pemahaman mendalam tentang topik-topik dari sumber makalah dan jurnal yang terkait dengan Oracle dan MongoDB. Pada tahap selanjutnya yaitu tahap identifikasi *software* dan *hardware* yang dimana pada tahap ini perlu melakukan identifikasi perangkat lunak dan perangkat keras yang akan digunakan sebagai media untuk pengujian ini karena hal ini dapat memengaruhi performa dari pemrosesan eksekusi *database*.

Setelah mengidentifikasi *software* dan *hardware* yang akan digunakan, langkah berikutnya adalah penentuan *dataset*. *Dataset* adalah kumpulan data terstruktur yang digunakan untuk melakukan analisis, penelitian atau pengembangan. Pada penelitian ini *dataset* yang digunakan adalah data-data seputar penjualan di suatu toko buku yaitu *database gravity_book* dari lima belas tabel atau objek. Pada penelitian ini menggunakan *database* yang tidak berelasi karena MongoDB merupakan salah satu *database* NoSQL. *Database* NoSQL atau non-relasional merupakan jenis *database* yang tidak menggunakan relasi pada tabelnya. Agar memperoleh hasil yang lebih akurat, pengujian ini hanya menggunakan satu tabel dari kedua jenis *database* tanpa menggunakan relasi pada tabel di SQL DB Oracle maupun MongoDB.

Tahap terakhir yaitu proses eksekusi kueri. Pada tahap ini, dilakukan pengujian dengan melakukan eksekusi terhadap kueri dengan memasukkan perintah-perintah operasi *database* antara lain *select*, *update*, *insert*, dan *delete*. Pada pengujian ini, dilakukan pengujian kueri dengan menggunakan batasan (*limit*) yang berbeda yaitu 750, 1500, 2250, dan 3000. Variasi batasan jumlah data yang diambil bertujuan untuk menganalisa performa kedua *database* tersebut. Setiap pemrosesan atau eksekusi kueri hasil *response time* akan dicatat dan dianalisa untuk membandingkan mana performa dari kedua *database* yang paling efisiensi.

HASIL DAN PEMBAHASAN

Identifikasi *Software* dan *Hardware*

Pada penelitian ini melakukan identifikasi perangkat lunak dan perangkat keras yang digunakan sebagai media untuk pengujian ini. Penentuan spesifikasi perangkat lunak dan perangkat keras ini sangat penting karena untuk memastikan hasil pemrosesan kueri dan validitas yang dihasilkan akurat. Berikut yang terlampir Tabel 1. dan Tabel 2. Spesifikasi perangkat lunak dan perangkat keras sebagai pendukung media yang digunakan.

Tabel 1. Spesifikasi *Software* Pendukung

No.	<i>Software</i>	Versi
1.	Oracle	10g
2.	MongoDB	4.4
3.	DataGrip	2023.1.2

Tabel 2. Spesifikasi *Hardware* Pendukung

No.	<i>Hardware</i>	Versi
1.	Memori	4GB DDR3
2.	Prosesor	Intel Core i5-4200U (1.60 GHz) DDR3
3.	Storage	SSD 256GB

Penentuan *Dataset*

Pada Penelitian ini menggunakan *dataset* dari *database gravity_book* menggunakan salah satu tabel yaitu *cust_order* dengan jumlah data sebanyak 352.000 baris. *Dataset* ini dipilih karena memiliki variasi data yang mempresentasikan data transaksi pelanggan pada sistem toko buku. Struktur tabel ini relative sederhana yang mencerminkan data transaksi bisnis dalam kondisi nyata. Dengan menggunakan *dataset* yang sama untuk kedua *database* dapat dilakukan perbandingan yang lebih objektif sehingga dapat dianalisis. Berikut *dataset* yang digunakan dalam *database gravity_book* yang menggunakan tabel *cust_order* di bawah ini.

Table Name	Engine	Auto Increment	Data Length	Description
address	InnoDB	0	96K	
address_status	InnoDB	0	16K	
author	InnoDB	0	400K	
book	InnoDB	0	1.5M	
book_author	InnoDB	0	1.5M	
book_language	InnoDB	0	16K	
country	InnoDB	0	16K	
cust_order	InnoDB	11,435	352K	
customer	InnoDB	0	160K	
customer_address	InnoDB	0	160K	
order_history	InnoDB	25,284	1.5M	
order_line	InnoDB	41,513	1.5M	
order_status	InnoDB	0	16K	
publisher	InnoDB	0	128K	
shipping_method	InnoDB	0	16K	

Gambar 2. Dataset Database gravity_book Tabel cust_order

Eksekusi Kueri

Pada tahap ini, dilakukan pengujian terhadap kueri yaitu *select*, *update*, *insert*, dan *delete*. Pada pengujian ini, dilakukan pengujian kueri dengan menggunakan batasan (*limit*) yang berbeda yaitu 750, 1500, 2250, dan 3000. Kueri pertama yang diuji adalah perintah *select* (membaca data) untuk kueri pengujian terlampir pada tabel berikut :

Tabel 3. Kueri Perintah *Select Data*

Database	Kueri
Oracle	<code>select * from cust_order where ROWNUM <= 750;</code>
MongoDB	<code>db.cust_order.find().limit(750);</code>

Kueri kedua yang dilakukan untuk pengujian adalah perintah *update* (memperbarui data) pada baris ke 750, 1500, 2250, dan 3000, seperti yang terlampir pada tabel berikut :

Tabel 4. Kueri Perintah *Update Data*

Database	Kueri
Oracle	<code>UPDATE cust_order</code> <code>SET</code> <code>order_id = 750,</code> <code>order_date = TO_DATE('2023-06-09 09:41:30',</code> <code>'YYYY-MM-DD HH24:MI:SS'),</code> <code>customer_id = 7552,</code> <code>shipping_method_id = 7552,</code> <code>dest_address_id = 7552</code> <code>WHERE order_id = '750';</code>
MongoDB	<code>db.customer_order.updateMany(</code> <code>{ order_id: '750' },</code> <code>{</code> <code> \$set: {</code> <code> order_id: 7552,</code>

```

        order_date: new Date("2023-06-09T09:41:30"),
        customer_id: 7552,
        shipping_method_id: 7552,
        dest_address_id: 7552
    }
}
);

```

Kueri ketiga yang dilakukan pengujian yaitu perintah *insert* (memasukkan data baru) pada empat limit dengan dilakukan *looping insert* sebanyak 720, 1500, 2250, dan 3000. Untuk kueri pada pengujian terlampir pada tabel berikut :

Tabel 5. Kueri Perintah *Insert Data*

Database	Kueri
Oracle	<pre> CREATE OR REPLACE PROCEDURE insert_proc IS counter INT := 1; BEGIN WHILE counter <= 750 LOOP INSERT INTO cust_order(order_id, order_date, customer_id, shipping_method_id, dest_address_id) VALUES (7551, TO_DATE('2023-06-08 09:41:30', 'YYYY-MM-DD HH24:MI:SS'), 7551, 7551, 7551); counter := counter + 1; END LOOP; END; / </pre>
MongoDB	<pre> EXECUTE insert_proc; for (var x=1; x<3000; x++){ db.cust_order.insertOne({ order_id: 7554, order_date: new Date("2023-06-11T09:41:30"), customer_id: 7554, shipping_method_id: 7554, dest_address_id: 7554})} </pre>

Kueri keempat yang dilakukan adalah pengujian perintah *delete* (menghapus data) pada pada keempat limit yang telah ditentukan yaitu 750, 1500, 2250, dan 3000. Untuk kueri terlampir pada tabel berikut :

Tabel 6. Kueri Perintah *Select Data*

Database	Kueri
Oracle	DELETE FROM customer_order WHERE order_id = '7552';

MongoDB db.customer_order.deleteMany({ order_id: '7552' });

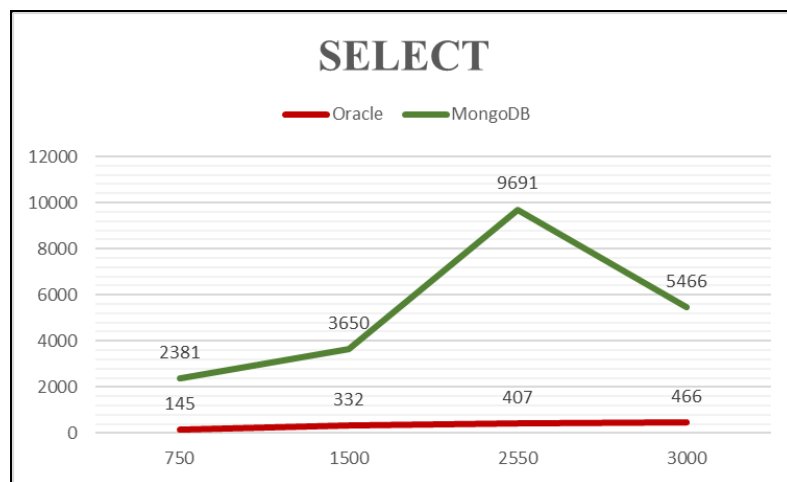
Berdasarkan metode penelitian yang telah dibuat, berikut pemaparan analisis pembahasan setelah eksekusi kueri select, update, insert, dan delete data dari kedua jenis DBMS yang berbeda yaitu Oracle dan MongoDB.

1. Select Data

Pada tahap pengujian perintah *select* memberikan hasil waktu respon kueri dengan menggunakan satuan *milisecond(ms)* yang ditampilkan dalam tabel grafik pengujian *select data* perbandingan antara SQL Oracle dan NoSql MongoDB seperti gambar dan tabel berikut :

Tabel 7. Hasil Pengujian Perintah *Select Data*

No.	Database	750	1500	2250	3000
1.	Oracle	145 ms	332 ms	407 ms	466 ms
2	MongoDB	2 s 381 ms	3 s 650 ms	9 s 691 ms	5 s 466 ms



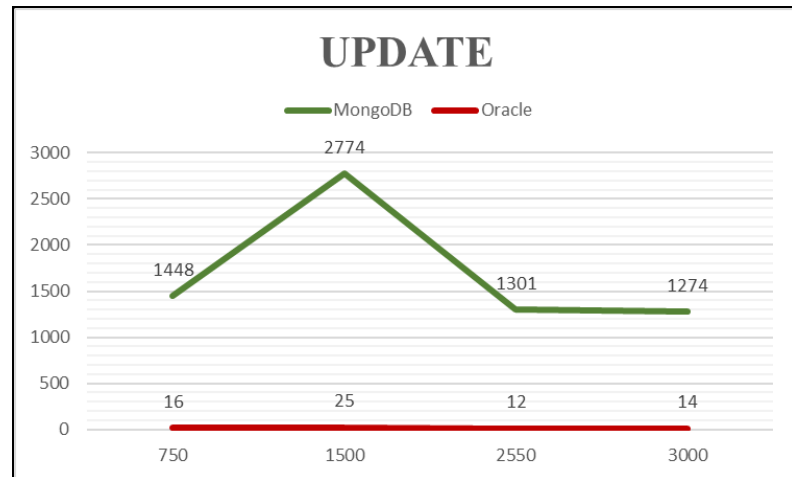
Gambar 3. Grafik Hasil Pengujian Perintah *Select Data*

2. Update Data

Pada tahap pengujian perintah *update* memberikan hasil waktu respon kueri dengan menggunakan satuan *milisecond(ms)* yang ditampilkan dalam tabel grafik pengujian *update data* perbandingan antara SQL Oracle dan NoSql MongoDB seperti gambar dan tabel berikut :

Tabel 8. Hasil Pengujian Perintah *Update Data*

No.	Database	750	1500	2250	3000
1.	Oracle	16 ms	25 ms	12 ms	14 ms
2	MongoDB	1 s 448 ms	2 s 774 ms	1 s 301 ms	1 s 274 ms



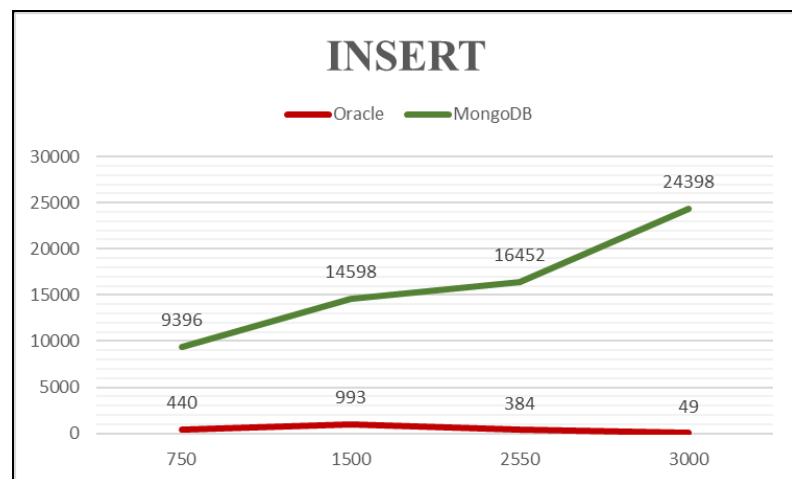
Gambar 4. Grafik Hasil Pengujian Perintah *Update Data*

3. *Insert Data*

Pada tahap pengujian perintah insert memberikan hasil waktu respon kueri dengan menggunakan satuan milisecond(ms) yang ditampilkan dalam tabel grafik pengujian insert data perbandingan antara Sql Oracle dan NoSql MongoDB seperti gambar dan tabel berikut :

Tabel 9. Hasil Pengujian Perintah *Insert Data*

No.	Database	750	1500	2250	3000
1.	Oracle	440 ms	993 ms	384 ms	49 ms
2	MongoDB	9 s 396 ms	14 s 598 ms	16 s 452 ms	24 s 398 ms



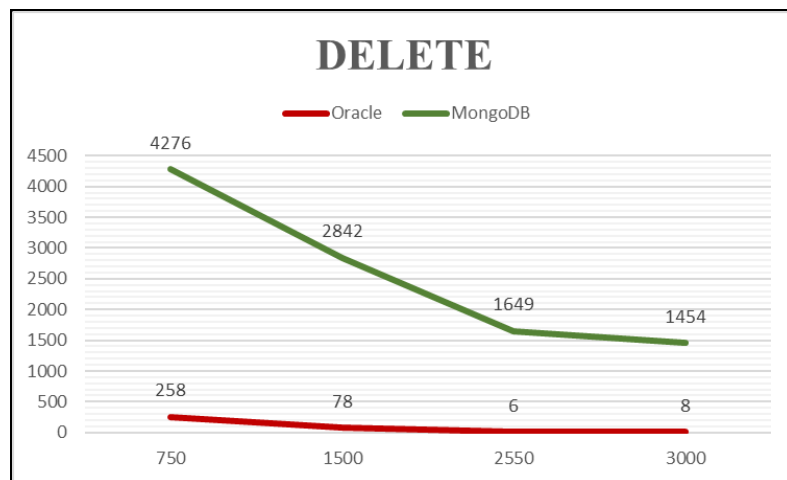
Gambar 5. Grafik Hasil Pengujian Perintah *Insert Data*

4. *Delete Data*

Pada tahap pengujian perintah *delete* memberikan hasil waktu respon kueri dengan menggunakan satuan *milisecond(ms)* yang ditampilkan dalam tabel grafik pengujian *delete* data perbandingan antara SQL Oracle dan NoSql MongoDB seperti gambar dan tabel berikut :

Tabel 10. Hasil Pengujian Perintah *Delete Data*

No.	Database	750	1500	2250	3000
1.	Oracle	258 ms	78 ms	6 ms	8 ms
2.	MongoDB	4 s 276 ms	2 s 842 ms	1 s 649 ms	1 s 458 ms

Gambar 6. Grafik Hasil Pengujian Perintah *Delete Data*

Berdasarkan semua pengujian kueri yang telah dilakukan untuk menganalisis perbandingan waktu respon kueri antara basis data SQL Oracle dan basis data NoSQL MongoDB, maka hasil pengujian telah diolah sehingga lebih mudah dipahami sebagai berikut:

Tabel 11. Hasil Semua Perintah Pengujian Kueri

Perintah	Oracle	MongoDB	Respond Time Tercepat
<i>SELECT</i>	1 s 350 ms	258 ms	Oracle
<i>UPDATE</i>	67 ms	4 s 276 ms	Oracle
<i>INSERT</i>	1 s 866 ms	64 s 844 ms	Oracle
<i>DELETE</i>	350 ms	10 s 221 ms	Oracle

Berdasarkan hasil seluruh pengujian yang telah dilakukan, dapat disimpulkan bahwa *respon time* tercepat untuk seluruh perintah pengujian kueri adalah Oracle, sedangkan MongoDB menempati posisi urutan kedua setelah Oracle.

KESIMPULAN

Dari pengujian ini mendapatkan hasil yang bisa disimpulkan bahwa pengujian pada basis data SQL Oracle menampilkan *response time* lebih unggul pada perintah *query* yang diujikan dibanding basis data NoSQL MongoDB. Dari eksekusi keempat *query* dengan empat *limit* yang ditentukan, mendapatkan hasil lebih unggul pada perintah

UPDATE menggunakan SQL Oracle dengan hasil *response time query* 67ms. Pengujian ini membuktikan bahwa basis data NoSQL MongoDB memiliki performa cenderung lemah dibanding Oracle jika dilihat dari kecepatan *response time* ketika mengeksekusi *query*. Pada pengujian berikutnya, disarankan untuk melakukan pengujian serupa dengan menggunakan basis data dan sistem operasi yang berbeda, serta menggunakan perintah *query* yang lebih kompleks.

DAFTAR PUSTAKA

- DB-Engines Ranking. (2025, 10 8). Retrieved from DB-ENGINES: <https://db-engines.com/en/ranking>
- Fadli, A., Zulfa, M. I., Nugraha, A. W., Taryana, A., & Aliim, M. S. (2020). Analisis Perbandingan Unjuk Kerja Database SQL dan Database NoSQL Untuk Mendukung Era Big Data. *Jurnal Nasional Teknik Elektro*.
- Hartanto, B., Ardiansyah, A., & Kurniasih, M. (2024). PENGENALAN DATABASE NOSQL DAN PERBANDINGANNYA DENGAN DATABASE RELASIONAL. *IPSIKOM*.
- Hartono, N., & Masyhur, Z. (2023). Optimizing transaction data performance in database management systems. *Matrix : Jurnal Manajemen Teknologi Dan Informatika*.
- Maukar, A. L., Marisa, F., Vitianingsih, A. V., & Puspitarini, E. W. (2022). *MODEL QUERY REALTIME DATABASE*. President University.
- Okman, L., Gal-Oz, N., Gonen, Y., Gudes, E., & Abramov, J. (2011). Security Issues in NoSQL Databases. *2011IEEE 10th International Conference on Trust*.
- Selviana, R., & Roji, M. F. (2023). *SISTEM BASIS DATA PART 1*. Tahta Media.
- Tavares, Inacio, O. M., Rangkoly, S. M., & Sarah Bunda Desy Bawan, E. U. (2020). Analisis Perbandingan Performansi Waktu Respons Kueri antara MySQL PHP 7.2. 27 dan NoSQL MongoDB. *Jurnal Teknologi Informasi*.
- Zhang, H., Chen, G., Ooi, B. C., Tan, K.-L., & Zhang, M. (2015). In-Memory Big Data Management and Processing: A Survey. *IEEE Transactions on Knowledge and Data Engineering*.