

Strategi *Vertical Scaling* dalam Meningkatkan Efisiensi Operasional Sistem Informasi Berbasis Database

Jan Everhard Riwurohi¹⁾, Arimaya Setyorini²⁾, Raden Bagus Dhana Pradana Adi³⁾, Tety Sapriani⁴⁾

¹⁾²⁾³⁾⁴⁾ Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Budi Luhur Jakarta
Jl. Ciledug Raya, Pertukangan Utara, Jakarta Selatan

¹⁾ yan.everhard@budiluhur.ac.id

²⁾ arimayasetyorini@gmail.com

³⁾ radenbagusdhana@gmail.com

⁴⁾ tetysapriani13@gmail.com

Abstrak

Efisiensi operasional sistem informasi sangat bergantung pada kinerja infrastruktur teknologi yang mendukungnya, khususnya server database yang berfungsi sebagai pusat pemrosesan dan penyimpanan data. Seiring meningkatnya jumlah pengguna dan volume transaksi, keterbatasan kapasitas sumber daya komputasi dapat menyebabkan penurunan performa sistem, meningkatnya waktu respons, serta potensi terjadinya *bottleneck* pada server database. Salah satu pendekatan yang dapat diterapkan untuk mengatasi permasalahan tersebut adalah *vertical scaling*, yaitu peningkatan kapasitas perangkat keras seperti CPU dan memori pada server yang telah ada. Penelitian ini bertujuan mengevaluasi dampak penerapan *vertical scaling* terhadap performa operasional sistem informasi berbasis database melalui studi kasus implementasi pada lingkungan perusahaan. Evaluasi dilakukan menggunakan beberapa indikator performa utama, yaitu utilisasi CPU, Input/Output Operations per Second (IOPS), waktu respons query, dan tingkat *availability* sistem. Hasil penelitian menunjukkan bahwa penerapan *vertical scaling* mampu meningkatkan performa I/O sebesar 57,83%, menurunkan beban CPU secara signifikan sebesar 10% hingga 40%, serta mempercepat waktu eksekusi query hingga 69,67%. Selain itu, stabilitas sistem juga meningkat yang ditunjukkan oleh peningkatan *availability* layanan. Temuan ini menunjukkan bahwa *vertical scaling* merupakan strategi yang efektif untuk meningkatkan efisiensi operasional sistem informasi berbasis database tanpa memerlukan perubahan arsitektur sistem secara menyeluruh.

Kata kunci: *cloud computing*, efisiensi operasional, performa database, sistem informasi, *vertical scaling*

Abstract

Operational efficiency in information systems largely depends on the performance of the underlying technology infrastructure, particularly database servers that function as the central components for data processing and storage. As the number of users and transaction volumes increase, limitations in computational resource capacity may lead to performance degradation, increased response time, and potential bottlenecks in the database server. One approach that can be applied to address this issue is vertical scaling, which involves increasing hardware resources such as CPU and memory on the existing server. This study aims to evaluate the impact of implementing vertical scaling on the operational performance of a database-based information system through a case study conducted in a corporate environment. The evaluation was performed using several key performance indicators, including CPU utilization, Input/Output Operations per Second (IOPS), query response time, and system availability. The results show that the implementation of vertical scaling improved I/O performance by 57.83%, significantly reduced CPU load by 10% to 40%, and accelerated query execution time by up to 69.67%. In addition, system stability also improved, as indicated by an increase in service availability. These findings demonstrate that vertical scaling is an effective strategy for improving the operational efficiency of database-based information systems without requiring major changes to the overall system architecture.

Keywords: *cloud computing*, operational efficiency, database performance, information systems, *vertical scaling*

1. PENDAHULUAN

Sistem informasi merupakan salah satu komponen penting dalam mendukung operasional organisasi modern. Sistem informasi dan teknologi informasi memiliki korelasi positif terhadap kinerja perusahaan karena mampu meningkatkan efisiensi dan efektivitas proses bisnis [1]. Berbagai aktivitas organisasi seperti pengelolaan keuangan, sumber daya manusia, penjualan, pelayanan pelanggan, hingga pengolahan data operasional dijalankan melalui sistem informasi yang terintegrasi. Secara konseptual, sistem informasi merupakan kombinasi antara teknologi, manusia, dan proses yang digunakan untuk mengumpulkan, mengolah, serta menyajikan informasi guna mendukung pengambilan keputusan dalam organisasi [2].

Dalam operasional sistem informasi, performa dan stabilitas sistem menjadi faktor yang sangat penting karena berpengaruh langsung terhadap produktivitas pengguna dan kualitas layanan yang diberikan kepada organisasi. Salah satu komponen utama yang menentukan performa sistem adalah server database, yang berfungsi sebagai pusat penyimpanan, pengolahan, dan distribusi data bagi berbagai aplikasi yang digunakan oleh organisasi [3]. Kinerja server database sangat bergantung pada kemampuan sumber daya komputasi, khususnya CPU dan memori, dalam menangani proses komputasi dan transaksi data secara efisien [4].

Seiring dengan meningkatnya jumlah pengguna dan volume data yang dikelola, beban kerja pada server database cenderung meningkat secara signifikan. Kondisi ini dapat menyebabkan penurunan performa sistem, meningkatnya waktu respons, serta potensi terjadinya gangguan layanan yang berdampak pada operasional organisasi. Apabila tidak ditangani dengan baik, keterbatasan kapasitas server dapat menjadi *bottleneck* yang menghambat efisiensi sistem informasi secara keseluruhan [5].

Untuk mengatasi permasalahan tersebut, berbagai pendekatan peningkatan kapasitas sistem telah dikembangkan, salah satunya melalui strategi *scaling*. Secara umum terdapat dua pendekatan utama, yaitu *horizontal scaling (scale out)* dan *vertical scaling (scale up)*. *Horizontal scaling* dilakukan dengan menambah jumlah server untuk mendistribusikan beban kerja ke beberapa node secara paralel, sehingga sering digunakan dalam sistem terdistribusi dan arsitektur cloud modern [6]. Namun pendekatan ini biasanya memerlukan perubahan arsitektur aplikasi, manajemen data yang lebih kompleks, serta mekanisme sinkronisasi antar server.

Sebaliknya, *vertical scaling* dilakukan dengan meningkatkan kapasitas perangkat keras pada server yang sudah ada, seperti penambahan jumlah *core* CPU, kapasitas RAM, maupun performa penyimpanan. Pendekatan ini relatif lebih sederhana karena tidak memerlukan perubahan signifikan pada arsitektur sistem yang telah berjalan. *Vertical scaling* banyak diterapkan pada sistem informasi berbasis database yang masih menggunakan arsitektur monolitik atau belum dirancang untuk distribusi beban kerja secara horizontal [6].

Dari sudut pandang pengelolaan sistem informasi, *vertical scaling* tidak hanya berkaitan dengan peningkatan spesifikasi perangkat keras, tetapi juga merupakan strategi untuk meningkatkan efisiensi operasional sistem. Dengan kapasitas komputasi yang lebih besar, sistem dapat menangani lebih banyak permintaan secara simultan, mengurangi waktu tunggu pengguna, serta meningkatkan stabilitas layanan teknologi informasi [7]. Oleh karena itu, pemantauan performa sistem melalui indikator seperti utilisasi CPU, memori, serta IOPS (*Input/Output Operations per Second*) menjadi penting dalam menentukan kebutuhan peningkatan kapasitas server.

Meskipun berbagai penelitian telah membahas strategi *scaling* dalam sistem komputasi, sebagian besar studi lebih banyak berfokus pada aspek arsitektur cloud atau sistem terdistribusi. Penelitian yang mengkaji secara langsung dampak implementasi *vertical scaling* terhadap performa operasional sistem informasi berbasis database dalam lingkungan organisasi nyata masih relatif terbatas. Selain itu, evaluasi performa seringkali hanya menggunakan satu atau dua indikator teknis, sehingga belum memberikan gambaran yang komprehensif mengenai dampak *scaling* terhadap stabilitas dan efisiensi sistem.

Oleh karena itu, penelitian ini bertujuan untuk mengevaluasi pengaruh implementasi *vertical scaling* terhadap performa sistem informasi berbasis database melalui studi kasus pada lingkungan operasional perusahaan. Evaluasi dilakukan secara komprehensif dengan menganalisis beberapa indikator performa utama, yaitu utilisasi CPU, IOPS, waktu respons query,

serta tingkat *availability* sistem. Dengan menggunakan pendekatan studi kasus pada lingkungan operasional nyata, penelitian ini diharapkan dapat memberikan kontribusi praktis sekaligus memperkaya kajian mengenai strategi optimalisasi performa sistem informasi berbasis database.

2. TINJAUAN PUSTAKA

2.1 Sistem Informasi Berbasis Database

Sistem informasi berbasis database merupakan sistem yang mengelola data dan informasi melalui mekanisme penyimpanan, manipulasi, dan pengambilan data yang terstruktur. Komponen utama sistem ini meliputi *database management system* (DBMS), server, jaringan, dan aplikasi client. DBMS digunakan untuk menangani transaksi data secara efisien dan andal. Sistem informasi memiliki komponen utama seperti perangkat keras, perangkat lunak, data, prosedur, dan pengguna yang saling berinteraksi untuk menghasilkan informasi bernilai guna pengambilan keputusan [8], [9]. Infrastruktur TI seperti server database memegang peranan penting dalam menjaga kecepatan, stabilitas, dan keandalan sistem informasi.

2.2 Vertical Scaling

Skalabilitas dalam sistem informasi merujuk pada kemampuan sistem untuk menangani peningkatan beban kerja secara efisien. Salah satu pendekatan yang biasa digunakan yaitu *vertical scaling* yang merupakan proses peningkatan kapasitas komputasi pada satu server dengan cara menambah *resource* seperti CPU, RAM, dan *storage*. Pendekatan ini banyak diterapkan pada sistem database yang belum didesain untuk distribusi horizontal [6]. *Vertical scaling* juga dikenal sebagai solusi cepat dan efisien dalam mendukung peningkatan performa sistem secara langsung [7]. Skalabilitas vertikal lebih mudah diterapkan pada sistem monolitik atau sistem yang membutuhkan konsistensi tinggi. Indikator performa sistem penyimpanan yang menunjukkan jumlah operasi baca dan tulis per detik (IOPS) dapat menandakan bagaimana server mampu menangani permintaan data dalam jumlah besar secara efisien, menjadi parameter penting dalam mengevaluasi performa sistem terutama saat beban kerja meningkat [10], [11].

3. METODE PENELITIAN

Pengujian dilakukan dalam periode waktu yang sama sebelum dan sesudah penerapan *vertical scaling*, masing-masing selama 7 hari operasional sistem, pada kondisi operasional normal dengan beban kerja yang relatif konstan, guna memastikan konsistensi beban kerja yang diuji dan merepresentasikan kondisi nyata sistem. Jumlah transaksi *dummy* disesuaikan dengan rata-rata volume transaksi harian sistem sehingga skenario pengujian mencerminkan aktivitas operasional perusahaan. Validasi hasil dilakukan dengan membandingkan nilai rata-rata dan tren perubahan setiap parameter performa pada kondisi sebelum dan sesudah *scaling*.

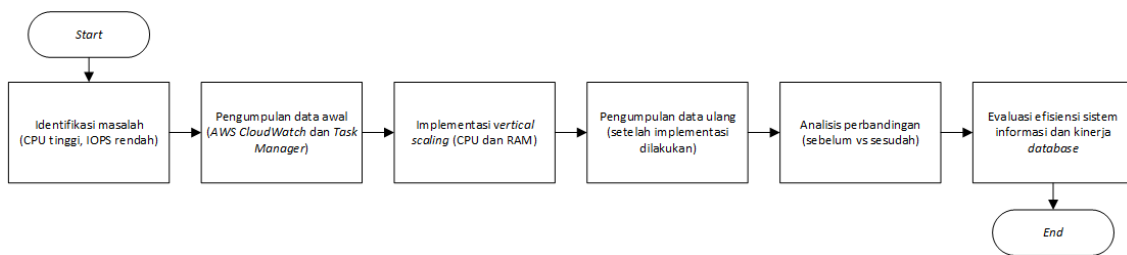
Data performa dikumpulkan sebelum dan sesudah dilakukan *vertical scaling* dan dianalisis secara komparatif menggunakan analisis presentase perubahan antar parameter. Analisis dilakukan untuk melihat adanya perbaikan secara signifikan atau tidak pada indikator performa sistem, serta sejauh mana *vertical scaling* dapat mendukung efisiensi operasional tanpa perubahan arsitektur sistem. Seluruh proses pengujian dilakukan dalam satu sesi eksperimen yang terkontrol untuk menjaga konsistensi dan hasil sesuai yang diharapkan.

Penelitian ini menggunakan pendekatan kuasi eksperimen untuk mengukur dampak *vertical scaling* terhadap performa sistem informasi berbasis database. Proses penelitian dilakukan dalam tiga tahap utama, yaitu perencanaan konfigurasi, pelaksanaan pengujian, dan analisis hasil.

Sumber data yang digunakan berasal dari sistem *monitoring* infrastruktur cloud melalui layanan *Amazon Web Service* (AWS), khususnya *Amazon CloudWatch*. Pengujian dilakukan pada lingkungan sistem berbasis *relational database management system* (RDBMS) yang berjalan di dalam *EC2 instance* dengan konfigurasi awal dan sesudah *scaling* berbeda pada jumlah CPU, kapasitas RAM, dan IOPS.

Teknik pengumpulan data dilakukan secara langsung melalui pengamatan terhadap parameter performa sistem menggunakan *CloudWatch* serta bantuan *Task Manager* pada *Windows Server*

dan *log database* internal. Parameter performa yang diamati meliputi *CPU utilization*, *memory usage*, *disk IOPS*, serta *query response time*. Untuk pengujian beban, digunakan simulasi aktivitas pengguna menggunakan data *dummy* yang mencerminkan pola transaksi rutin dalam sistem informasi perusahaan seperti eksekusi query, pembaruan data, dan permintaan laporan dengan keseluruhan jumlah *row* data yang digunakan adalah 684 rows dengan adanya 15 *background job* yang sedang berjalan pada kondisi operasional *business as usual* dengan *concurrent user* yang mengakses sebanyak 20 user. Data performa dikumpulkan sebelum dan sesudah dilakukan *vertical scaling* dan dianalisis secara komparatif menggunakan analisis persentase perubahan antar parameter. Data performa yang diambil berupa kemampuan berapa waktu dalam satuan menit yang dibutuhkan server untuk mengeksekusi query yang nantinya akan menghasilkan 684 rows data disaat pemakaian server berjalan pada posisi operasional.



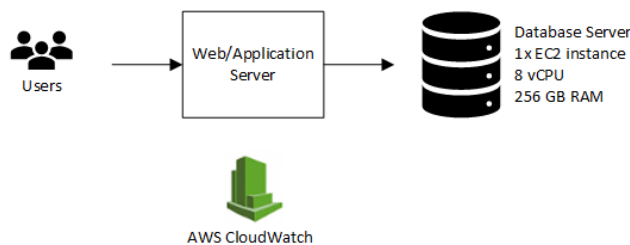
Gambar 1. Alur proses penelitian *vertical scaling*

Analisis dilakukan untuk melihat adanya perbaikan secara signifikan atau tidak pada indikator performa sistem, serta sejauh mana *vertical scaling* dapat mendukung efisiensi operasional tanpa perubahan arsitektur sistem. Seluruh proses pengujian dilakukan dalam satu sesi eksperimen yang terkontrol untuk menjaga konsistensi dan hasil sesuai yang diharapkan.

4. PEMBAHASAN

Bagian ini menyajikan hasil implementasi *vertical scaling* pada sistem informasi berbasis database, serta analisis perubahan performa sebelum dan sesudah peningkatan kapasitas server. Parameter utama yang diamati meliputi utilisasi CPU dan IOPS (*Input/Output Operations per Second*), *query response time*, dan tingkat *availability* sistem, karena keempat indikator tersebut merepresentasikan kinerja komputasi, kemampuan pemrosesan transaksi, serta stabilitas operasional sistem.

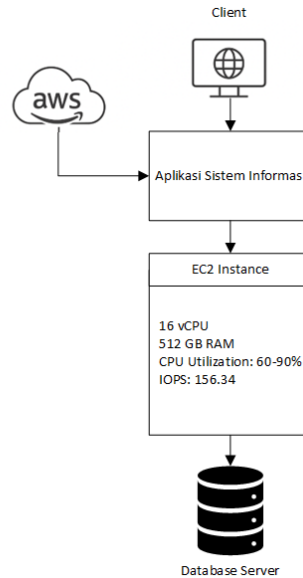
Implementasi dilakukan pada infrastruktur cloud menggunakan layanan AWS yang menjalankan sistem database utama perusahaan. Pada kondisi awal, sistem berjalan pada satu instance EC2 dengan spesifikasi 8 vCPU dan 256 GB RAM (Gambar 2).



Gambar 2. Arsitektur sistem sebelum *vertical scaling*

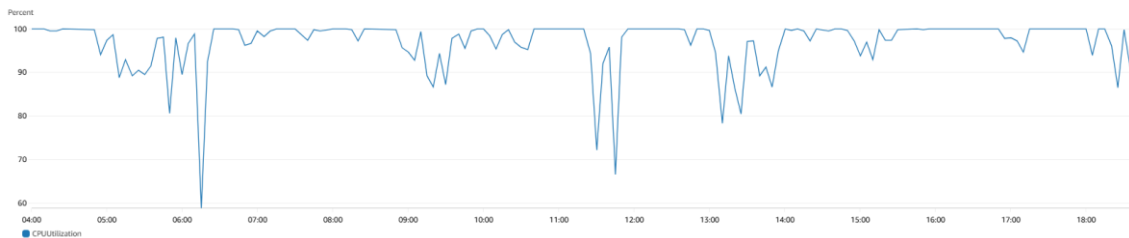
Setelah dilakukan *vertical scaling*, kapasitas *instance* ditingkatkan menjadi 16 vCPU dan 512 GB RAM tanpa perubahan pada arsitektur aplikasi maupun struktur sistem (Gambar 3). Pendekatan ini menunjukkan bahwa peningkatan kapasitas komputasi dapat dilakukan tanpa memerlukan perubahan desain sistem secara menyeluruh. Sebelum proses *scaling* dilakukan, sistem terlebih dahulu melalui proses *backup* untuk memastikan ketersediaan mekanisme pemulihan apabila terjadi kegagalan

selama proses peningkatan kapasitas. *Instance* kemudian dihentikan sementara untuk melakukan proses peningkatan spesifikasi, setelah itu sistem diaktifkan kembali dengan konfigurasi *services* dan *scheduler* yang sama seperti sebelumnya. Pemantauan performa dilakukan pada periode operasional yang sebanding guna memperoleh data yang representatif sebelum dan sesudah implementasi *scaling*.



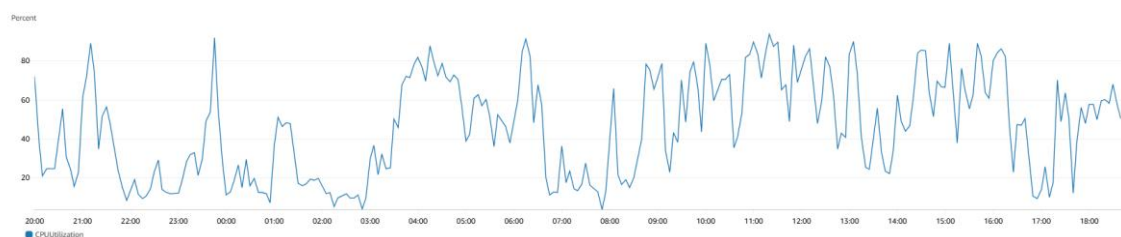
Gambar 3. Arsitektur sistem setelah *vertical scaling*

Hasil monitoring menunjukkan bahwa sebelum penerapan *vertical scaling*, utilisasi CPU berada pada kondisi mendekati 100% yang terindikasi *full load* secara konsisten selama jam operasional (Gambar 4). Kondisi ini mengindikasikan bahwa server telah mencapai batas kapasitas pemrosesan dan berpotensi menjadi *bottleneck* bagi sistem informasi.



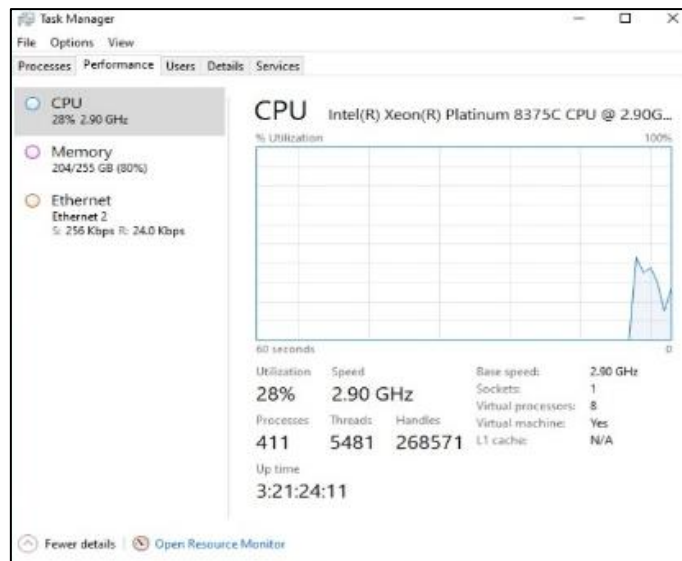
Gambar 4. Grafik CPU *utilization* dari AWS CloudWatch sebelum *scaling*

Setelah peningkatan kapasitas dilakukan, grafik utilisasi CPU menunjukkan pola fluktuasi yang lebih stabil pada kisaran 60–90% (Gambar 5). Perubahan ini menandakan bahwa sistem memiliki ruang pemrosesan tambahan (*processing headroom*) yang memungkinkan penanganan beban kerja secara lebih efisien.



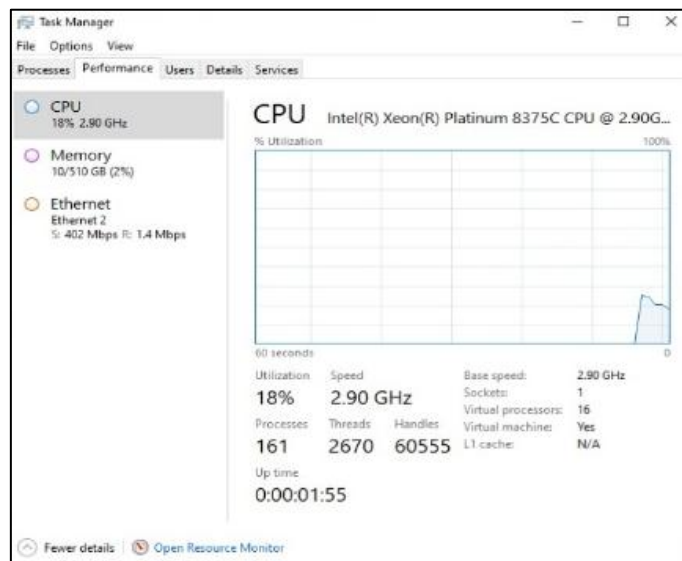
Gambar 5. Grafik CPU *utilization* dari AWS CloudWatch setelah *scaling*

Jika dilihat dari sisi *task manager* pada windows server, terlihat pada gambar 6 bahwa jumlah *logical processor* yang tersedia hanya sebanyak 8 *core* dengan RAM 256 GB.



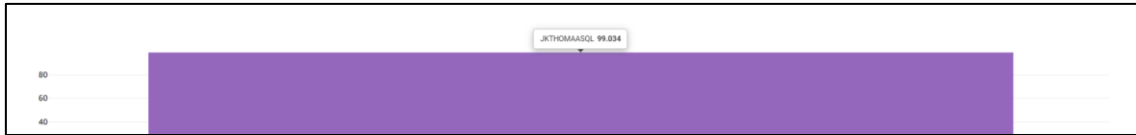
Gambar 6. Informasi dari *task manager* sebelum *scaling*

Setelah *scaling* diterapkan, jumlah *logical processor* bertambah menjadi 16 *core* dengan kapasitas RAM bertambah menjadi 512 GB. Proses *scaling* berhasil meningkatkan sumber daya komputasi tanpa perubahan arsitektur sistem secara menyeluruh.

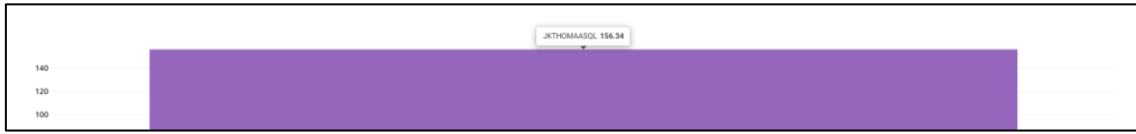


Gambar 7. Informasi dari *task manager* setelah *scaling*

Peningkatan kapasitas sumber daya juga terlihat dari sisi IOPS pada Gambar 8 dan 9 yang mengalami peningkatan dari 99.034 menjadi 156.34. Peningkatan ini menunjukkan bahwa sistem memiliki kemampuan yang lebih baik dalam menangani operasi baca dan tulis data pada database. Selain itu, pengujian terhadap waktu eksekusi query yang sama menunjukkan penurunan dari 28.25 detik menjadi 8.57 detik, yang berarti terjadi percepatan proses sebesar 69.67%. Hasil ini menunjukkan bahwa peningkatan kapasitas komputasi secara langsung berdampak pada peningkatan efisiensi pemrosesan transaksi database, khususnya pada sistem database monolitik [6].

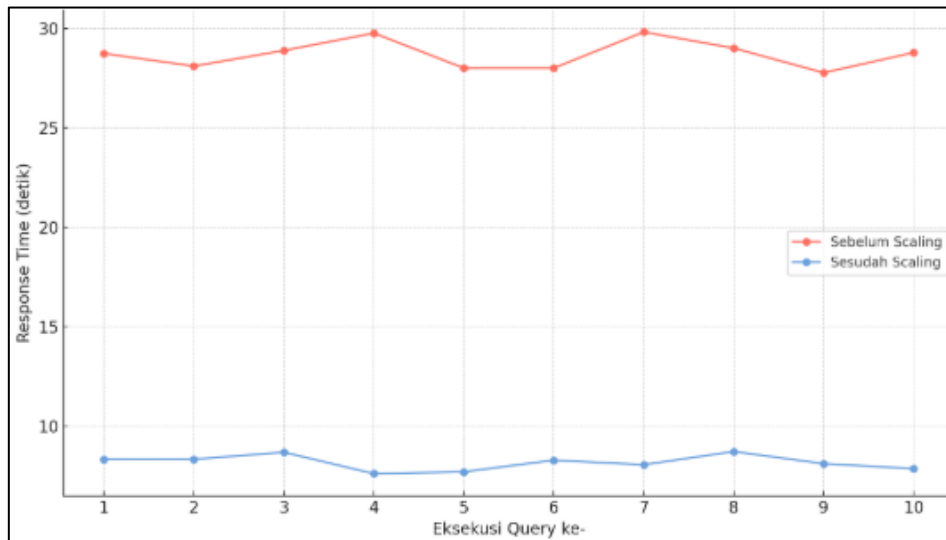


Gambar 8. IOPS sebelum *scaling*



Gambar 9. IOPS setelah *scaling*

Peningkatan performa juga terlihat dari pengujian waktu eksekusi query yang sama. Sebelum *vertical scaling*, waktu eksekusi mencapai 28.25 detik, sementara setelah *vertical scaling* hanya membutuhkan waktu 8.57 detik (Gambar 10). Hal ini berarti terjadi percepatan proses sebesar 69.67%, menunjukkan peningkatan efisiensi dalam pengolahan data pada sisi server database yang sangat signifikan.



Gambar 10. Perbandingan waktu eksekusi *query*

Secara keseluruhan, perbandingan performa sebelum dan sesudah *vertical scaling* ditunjukkan pada Tabel 1. Kapasitas CPU dan memori meningkat dua kali lipat, yang diikuti oleh peningkatan *throughput* I/O, penurunan waktu respons query, serta peningkatan *availability* sistem dari 98.50% menjadi 99.99%. Peningkatan ini menunjukkan bahwa ketersediaan sumber daya komputasi memiliki korelasi langsung terhadap stabilitas operasional sistem informasi berbasis database.

Tabel 1. Perbandingan performa server

Parameter	Sebelum	Sesudah	Perubahan
CPU	8 vCPU	16 vCPU	+100%
RAM	256 GB	512 GB	+100%
CPU Utilization (Avg)	100% full load	60 – 90% fluktuatif	-25 %
IOPS	99.034	156.34	+57.83%
Response time	28.25 detik	8.57 detik	-69.67 %
System availability	98.50%	99.99%	+1.51%

Peningkatan performa sistem setelah penerapan *vertical scaling* tidak semata-mata disebabkan oleh penambahan kapasitas perangkat keras, tetapi juga oleh berkurangnya *resource contention*

pada server database. Dalam konteks sistem yang dianalisis, server database tidak hanya menangani eksekusi query pengguna, tetapi juga menjalankan sejumlah *background job* terjadwal serta melayani akses pengguna secara simultan. Kondisi ini menciptakan pola beban kerja campuran (*mixed workload*) yang bersifat CPU bound dan I/O bound secara bersamaan.

Pada kondisi sebelum *vertical scaling*, keberadaan sekitar 10 *background job* yang berjalan paralel dengan permintaan dari sekitar 20 *concurrent user* menyebabkan kompetisi intensif terhadap sumber daya CPU. Sistem operasi harus membagi waktu pemrosesan ke berbagai proses aktif, sehingga setiap query memperoleh jatah pemrosesan yang terbatas. Akibatnya, latensi meningkat bukan semata karena kompleksitas query, tetapi juga akibat tingginya waktu tunggu (*waiting time*) dalam antrean eksekusi proses. Fenomena ini menjelaskan mengapa utilisasi CPU berada pada kondisi *full load* secara konsisten.

Setelah jumlah CPU ditingkatkan, sistem memiliki tingkat paralelisme pemrosesan yang lebih tinggi. Dengan bertambahnya *logical processor*, mekanisme proses *scheduling* pada sistem operasi dapat menjalankan lebih banyak proses secara simultan tanpa peningkatan *context switching* yang berlebihan. Kondisi ini mengurangi CPU *scheduling overhead* dan memungkinkan query pengguna serta *background job* dieksekusi secara lebih efisien.

Dari sisi I/O, peningkatan performa juga dipengaruhi oleh peningkatan kapasitas memori utama. Pada konfigurasi awal, keterbatasan memori menyebabkan tingginya frekuensi akses disk karena sebagian data tidak dapat dipertahankan dalam *buffer cache*. Setelah kapasitas RAM ditingkatkan, lebih banyak data dapat diproses secara *in memory*, sehingga tekanan terhadap sistem penyimpanan berkurang dan efektivitas penjadwalan I/O meningkat. Hal ini menjelaskan peningkatan nilai IOPS yang diamati setelah implementasi *vertical scaling*.

Penurunan waktu eksekusi query merupakan implikasi langsung dari peningkatan kapasitas CPU dan memori tersebut. Dalam sistem database dengan beban kerja paralel, waktu respons dipengaruhi oleh waktu pemrosesan aktual (*service time*) serta waktu tunggu akibat kompetisi sumber daya. *Vertical scaling* secara efektif menurunkan kedua komponen tersebut yaitu waktu pemrosesan berkurang karena peningkatan kapasitas komputasi, sementara waktu tunggu berkurang karena menurunnya *resource contention*.

Dari perspektif *availability*, kondisi sebelum *scaling* menunjukkan sistem beroperasi sangat dekat dengan batas kapasitasnya, sehingga lonjakan beban kecil sekalipun dapat menyebabkan penurunan performa atau gangguan layanan. Setelah *vertical scaling*, tersedianya ruang kapasitas tambahan memungkinkan sistem menyerap variasi beban kerja tanpa mengalami degradasi layanan secara signifikan.

Secara konseptual, temuan ini menunjukkan bahwa *vertical scaling* merupakan strategi optimalisasi yang efektif untuk sistem informasi berbasis database dengan arsitektur monolitik dan beban kerja paralel. Meskipun demikian, pendekatan ini memiliki keterbatasan karena seluruh komponen sistem tetap bergantung pada satu node dan kapasitas perangkat keras memiliki batas maksimum peningkatan. Oleh karena itu, *vertical scaling* dapat diposisikan sebagai strategi optimalisasi jangka pendek hingga menengah sebelum organisasi mempertimbangkan pendekatan lain seperti *horizontal scaling* atau arsitektur terdistribusi untuk mencapai skalabilitas yang lebih tinggi.

5. KESIMPULAN

Penelitian ini menunjukkan bahwa penerapan *vertical scaling* memberikan dampak signifikan terhadap peningkatan efisiensi operasional sistem informasi berbasis database. Berdasarkan hasil evaluasi, waktu eksekusi query menurun dari rata-rata 28,25 detik menjadi 8,57 detik setelah proses *scaling*, yang menunjukkan peningkatan efisiensi waktu respons sebesar 69,67%. Selain itu, performa *Input/Output Operations per Second* (IOPS) meningkat dari 99.034 menjadi 156.34, sementara utilisasi CPU yang sebelumnya berada pada kondisi penuh (100%) menjadi lebih stabil pada kisaran 60% hingga 90%. Peningkatan kapasitas CPU dan memori hingga dua kali lipat juga berdampak pada meningkatnya stabilitas sistem serta menurunnya potensi downtime, sehingga secara langsung meningkatkan kecepatan respons aplikasi dan pengalaman pengguna.

Hasil penelitian ini menunjukkan bahwa peningkatan performa sistem tidak semata-mata disebabkan oleh penambahan kapasitas perangkat keras, tetapi juga oleh berkurangnya *resource*

contention serta meningkatnya efisiensi pemrosesan paralel pada server database. Dalam lingkungan operasional dengan beban kerja paralel yang melibatkan proses latar belakang dan akses pengguna secara simultan, peningkatan kapasitas komputasi memungkinkan sistem menangani permintaan secara lebih efisien tanpa memerlukan perubahan arsitektur sistem secara menyeluruh.

Secara ilmiah, penelitian ini memberikan kontribusi pada bidang sistem informasi melalui evaluasi performa *vertical scaling* yang dilakukan secara komprehensif dengan mempertimbangkan beberapa indikator utama secara simultan, yaitu utilisasi CPU, IOPS, waktu respons, dan *availability* sistem dalam konteks implementasi pada lingkungan operasional nyata. Berbeda dengan sebagian besar penelitian sebelumnya yang mengevaluasi performa *scaling* pada lingkungan simulasi atau hanya menggunakan satu indikator kinerja, penelitian ini menganalisis dampak *vertical scaling* secara komprehensif melalui beberapa metrik performa sekaligus dalam konteks implementasi nyata sistem informasi perusahaan. Temuan ini memperkaya pemahaman mengenai efektivitas *vertical scaling* sebagai strategi optimasi performa sistem informasi berbasis database, khususnya pada sistem dengan arsitektur monolitik dan beban kerja paralel.

Meskipun demikian, *vertical scaling* memiliki keterbatasan terkait batas fisik kapasitas server dan ketergantungan seluruh komponen sistem pada satu node. Oleh karena itu, pendekatan ini lebih tepat diposisikan sebagai strategi optimasi jangka pendek hingga menengah sebelum organisasi mempertimbangkan pendekatan lain seperti *horizontal scaling* atau mekanisme *failover* untuk mencapai skalabilitas dan resiliensi sistem yang lebih tinggi.

DAFTAR PUSTAKA

- [1] F. Maira and M. I. P. Nasution, "Peranan serta implikasi sistem informasi manajemen dan teknologi informasi dalam mengoptimalkan kinerja bisnis suatu perusahaan," *Kohesi: Jurnal Sains dan Teknologi*, vol. 3, no. 1, pp. 20–29, 2024.
- [2] K. C. . Laudon and J. P. . Laudon, *Management information systems : managing the digital firm*. Pearson Education, 2020.
- [3] R. Behl, J. A. O'Brien, and G. M. Marakas, *Management information systems*. Noida: McGraw Hill (India), 2019.
- [4] Y. Cahyaningrum *et al.*, *Arsitektur dan organisasi komputer*. Yogyakarta: PT Penamuda Media, 2023.
- [5] N. Nadia and M. I. P. Nasution, "Transformasi digital: peran sistem informasi manajemen dalam meningkatkan efisiensi organisasi," *Jurnal Ilmiah Ekonomi, Manajemen, Bisnis dan Akuntansi*, vol. 2, no. 1, pp. 627–634, 2025, doi: 10.61722/jemba.v2i1.675.
- [6] U. K. Manne, "Horizontal vs. vertical scaling in modern database systems: a comparative analysis of performance and cost trade-offs," *International Journal of Computer Engineering and Technology (IJCET)*, vol. 15, no. 5, pp. 514–524, 2024, doi: 10.5281/zenodo.13851050.
- [7] J. Wang *et al.*, "PolarDB-IMCI: a cloud-native HTAP database system at Alibaba," *Proceedings of the ACM on Management of Data*, vol. 1, no. 2, pp. 1–25, Jun. 2023, doi: 10.1145/3589785.
- [8] Abraham. Silberschatz, H. F. Korth, and S. Sudarshan, *Database system concepts*. McGraw-Hill, 2020.
- [9] A. Asriani, M. Amir, and A. Kadir, *Sistem informasi manajemen teori & prinsip-prinsip dasar*. Kendari: CV. Literasi Indonesia, 2024.
- [10] N. Dragoni *et al.*, "Microservices: yesterday, today, and tomorrow," in *Present and Ulterior Software Engineering*, Cham: Springer International Publishing, 2017, pp. 195–216. doi: 10.1007/978-3-319-67425-4_12.
- [11] C. Pahl and P. Jamshidi, "Microservices: a systematic mapping study," in *Proceedings of the 6th International Conference on Cloud Computing and Services Science*, SCITEPRESS - Science and Technology Publications, 2016, pp. 137–146. doi: 10.5220/0005785501370146.

Biodata Penulis

Dr. Ir. Jan Everhard Riwurohi, M.T., merupakan dosen di Universitas Budi Luhur Jakarta. Menempuh pendidikan akhir pada bidang Instrumentasi dan Elektronika di Universitas Gajah Mada dan saat ini juga berprofesi sebagai konsultan teknik pada beberapa industri di Tangerang Banten.

Arimaya Setyorini, S.Kom, mahasiswi program studi Ilmu Komputer di Universitas Budi Luhur Jakarta. Berprofesi sebagai IT Leader dalam pengembangan sistem di sebuah perusahaan swasta yang berlokasi di Jakarta.

Raden Bagus Dhana Pradana Adi, S.Kom, mahasiswa program studi Ilmu Komputer di Universitas Budi Luhur Jakarta. Berkecimpung dalam berbagai proyek *IT Network and Infrastructure Architecture*, di sebuah perusahaan retail.

Tety Sapriani, S.Kom, mahasiswi program studi Ilmu Komputer di Universitas Budi Luhur Jakarta. Berprofesi sebagai operator sekolah, kaprog Teknik Komputer Jaringan dan pengajar sekaligus instruktur kursus di bidang ilmu komputer di sekolah menengah kejuruan.