AlloyDB, an option for Enterprise Workloads?

Table of contents

accenture Google Cloud

Table of contents

accenture Google Cloud

What is AlloyDB?

AlloyDB is a fully managed and scalable, PostgreSQL-compatible database created by Google Cloud.

Google has combined the open-source database with Google native services which decouples the compute and storage layer improving processing speeds and allowing scalability and high availability. AlloyDB is being added as a more scalable and performant option to the existing Cloud SQL for PostgreSQL offering. One of the key differentiators is the machine learning (ML) capabilities of AlloyDB. A columnar store engine is available providing additional speed benefits to analytics and in-database ML usage patterns.

Some of the key features Google notes for AlloyDB include:

- 99.99% availability (inclusive of maintenance)
- Automated administration tasks including backups, replication, patching, and capacity management
- Transaction processing 4X faster than standard PostgreSQL
- Up to 100X faster analytical queries than a standard PostgreSQL implementation with no impact on operational performance
- Powers generative AI workloads via AlloyDB AI and its native integration with GCP's Vertex AI platform
- Transparent and predictable pricing with no licensing fees

Automated Managed Service Benefits

AlloyDB handles database patching, backups, storage scaling, and replication as part of the managed service, allowing data specialist resources to focus on value-add tasks and not administration.

In addition to these standard managed service features, Google implemented ML and adaptive algorithms to manage PostgreSQL vacuum management, storage and memory management and data tiering. It learns your workloads and organizes your data across memory, cache, and durable storage based on your workloads. This further simplifies the administration of the database freeing up additional time for DBAs and developers to work on new features.

AlloyDB also offers the ability to enable a columnar engine for in-database analytics. It too is intelligent and can provide recommendations on configuration and column population based on queries that have been submitted. With this feature enabled, AlloyDB can provide support for both OLTP and OLAP workloads.

High Availability

A critical component of enterprise, mission critical applications is high availability. A key feature of AlloyDB is the disaggregation of the database layer and storage.

Google has introduced an intelligent storage service that has been optimized for PostgreSQL and consists of 3 main components: regional log storage service, log processing service, and block storage service.

Regional log storage service is a low-latency service for fast write-ahead log (WAL) writing reflecting all database modification operations.

The log processing service (LPS) provides asynchronous processing of the WAL that generates a new version of data blocks by replaying the contents in the WAL records. It then sends the data blocks to block storage service for persistence. It also serves these updated data blocks to the primary and replicas.

Block Storage Service - stores the data blocks across availability zones and sends relevant data blocks to the log processing service whenever needed.

This design allows for protection against zonal failures without impact or modifications to the database layer. The storage service handles backups and does not impact the performance and resources of the database layer. In addition to the backup provided by the storage layer, AlloyDB also offers manual and scheduled back-ups for application-level failures.

Disaster Recovery (DR)

Cross Region Replication (CRR) in AlloyDB.

AlloyDB's CRR feature allows for the establishment of secondary clusters and instances from a primary cluster, ensuring resource availability in multiple regions and providing a DR capability in the event of outage in the primary region. These secondary clusters act as replicas of the primary cluster and data is asynchronously replicated from the primary cluster via physical replication. In the event of a primary cluster failure, the secondary cluster can be promoted to function as the new primary.

Backup and Recovery

On-demand or scheduled back-ups can be taken of the cluster data and is stored in a location separate from the cluster's data storage.

Access to the back-ups is provided by resources separate from the cluster so that once the backup is complete, it can be restored even if the cluster has been deleted. Automatic backups allow you to specify the date and time backups are to be taken and a retention policy. AlloyDB also provides a Point in Time Recovery feature, enabling the database to be restored to a chosen point in time. At the time of this writing, back-ups can only be restored to the region they originated. Google has on its roadmap the capability to backup into another region than the original cluster's region.

AlloyDB performance benchmarks

Setup

Benchmarking for AlloyDB was performed using HammerDB. HammerDB implemented TPROC-C which is the OLTP workload derived from the TPC-C specification. The name, TPROC-C , means "Transaction Processing Benchmark derived from the TPC "C" specification". This derivation adheres to the TPC Fair Use rules. The TPC-C specification implements a computer system that manages orders for a company. The items the company sells is fixed at 100,000 and these items are stocked in warehouses. Each warehouse (fictional physical warehouse not data warehouse) has 10 sales districts and each district has 3000 customers. The number of warehouses is configurable and for the benchmarking run in this benchmarking experiment, 1000 warehouses were configured. Customer orders are input by operators and contain a number of items. The TPROC-C schema, illustrated in Figure 1, implementation of this computer system with the rows based on the configuration used in the following benchmarks of 1000 warehouses.

The results of the TPROC-C benchmark cannot be compared to published TPC-C results however, the TPROC-C results can be compared TPROC-C results on different databases. HammerDB produces 2 statistics to compare across systems, NOPM (new orders per minute) and TPM (transactions per minute). Because the NOPM value is based on a metric captured from with the test schema itself, it is independent of any database implementation and is the recommended metric. (HammerDB, 2022)

TPROC-C Schema

TPROC-C The database schema and transactions for the tests are based on a supply chain company that sells items and keeps it stock in warehouses. For the tests, the number of warehouses and the number of users can be configured (for additional details on the system and schema see the HammerDB documentation https://www.hammerdb.com/docs/ch03s05.html).

The configuration for the NOPM tests run had 1000 warehouses and the number of users were increased with each run in the following increments: 1, 16, 32, 64, 128, and 256.

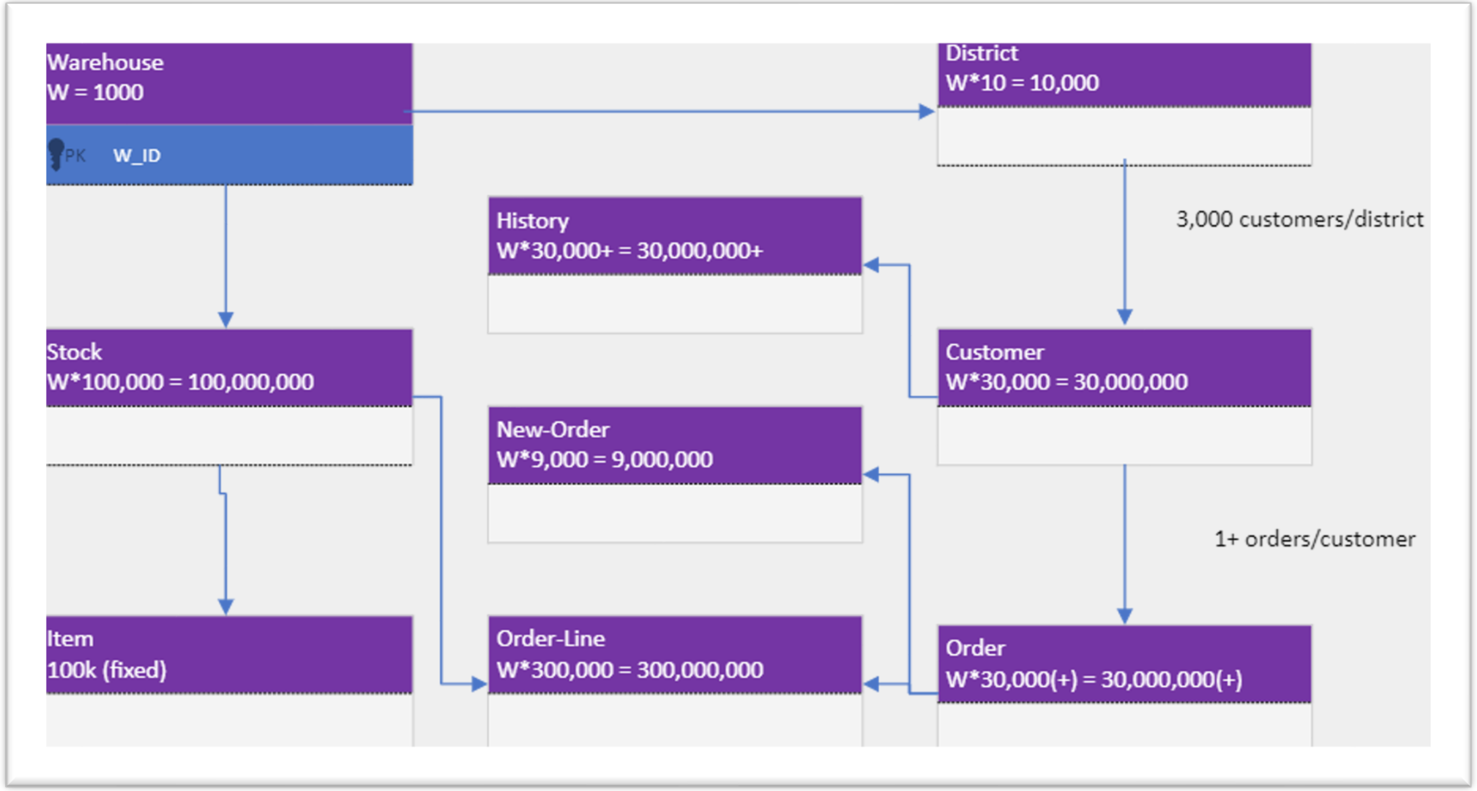


Figure 1 – TPROC-C database schema

The TPROC-C NOPM benchmark was run on the following databases:

OCI ATP (32 OCPU[1] Autonomous Memory v19c)
OCI MySQL Cloud Service (32 OCPUs 512GB Memory)
AWS Aurora PostgreSQL (64 vCPUs v13.4)
AlloyDB For PostgreSQL (64 vCPUs 512GB Memory v14)

[1] - 1 OCPU = 2vCPUs

Results

AlloyDB was able to process 20,164 new orders per minute with one user up to 841,035 new orders per minute with 256 users (Figure 2)

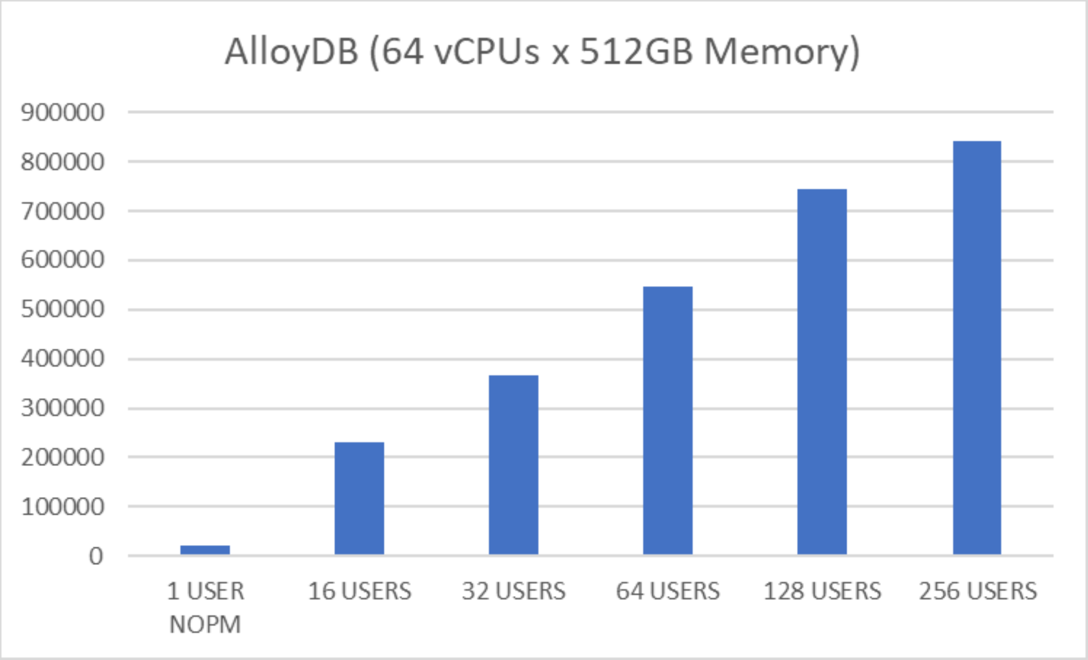


Figure 2 -AlloyDB New Order Per Minute transactions by user count

The NOPM transactions account for ~45% of the total transactions happening on the database. There are 5 transactions occurring on the database during the test:

- New-Order: new order received from customer (46%)
- Payment: recording a payment updating customer balance (45%)
- Delivery: asynchronous delivery of products (6%)
- Order status: lookup the status of the customers most recent order (2%)
- Stock level: lookup the status of warehouse inventory (1%)

For the 256 users, there were ~1,934,759 TPM of which 841,035 were NOPM.

The results in Figure 2 were an average of 3 runs and setup included the following setting:

- pg_rampup 1
- pg_duration 4
- pg_allwarehouse false

The TPROC-C test was also run against additional databases for comparison, and the NOPM for each of the databases are depicted in Figure 3.

OCI ATP did outperform AlloyDB by a small margin when 256+ users were hitting the system however, AlloyDB was still performant at 800,000+ NOPM at 256 users. While performance is a major consideration when selecting a database, cost is also another significant consideration.

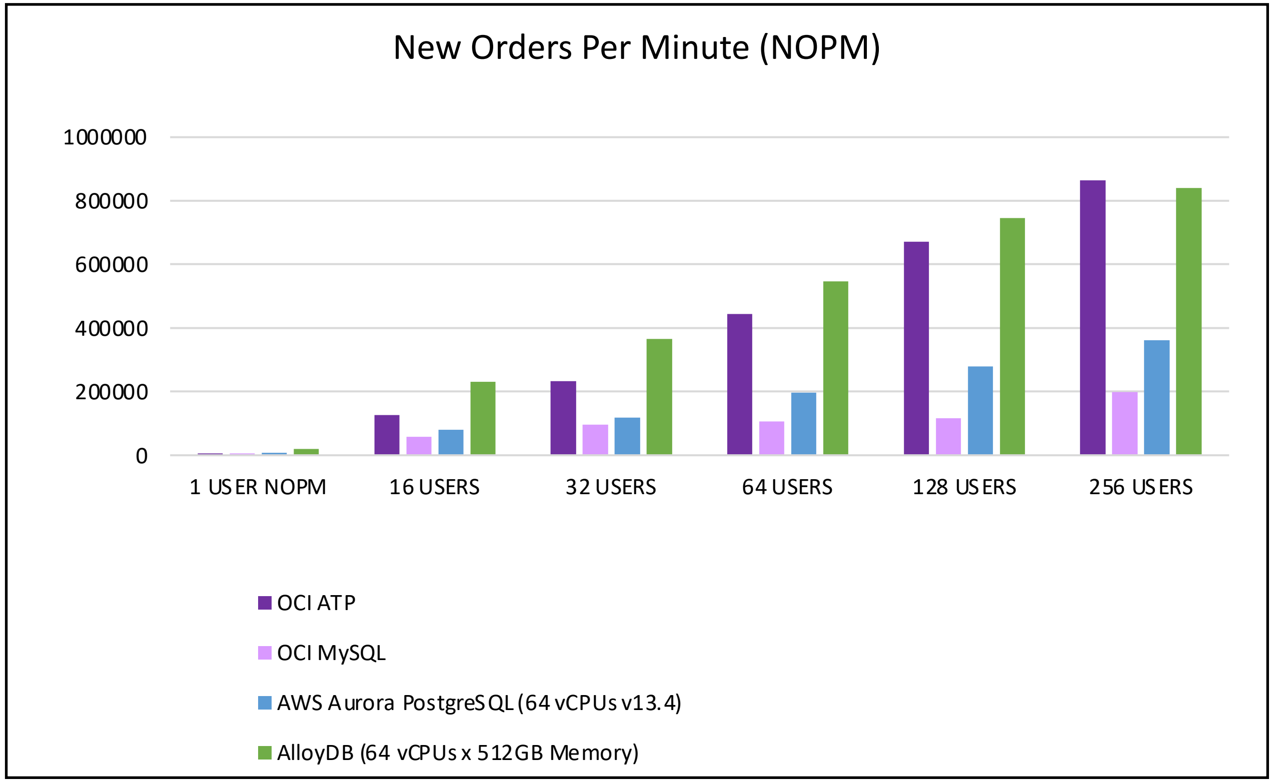


Figure 3 - New Orders Per Minute for cloud databases by user counts

**The results for all databases in Figure 3 except AlloyDB were shared with us for this POV from the Accenture Enkitec Group and will be included in another paper.*

Monthly Pricing by Database

Each cloud service provider provides pricing with different parameters. The data used for monthly pricing for the purpose of this analysis is below and details are in the appendix.

Several assumptions were made for pricing based on requirements from the CSP:

- For Aurora, IO estimates were based on benchmarking results
- For the Oracle ATP 99.995% HA SLA, Data Guard is required which also requires a secondary instance so the price was doubled per month to account for the standby. The price of Data Guard is not included in the table below.

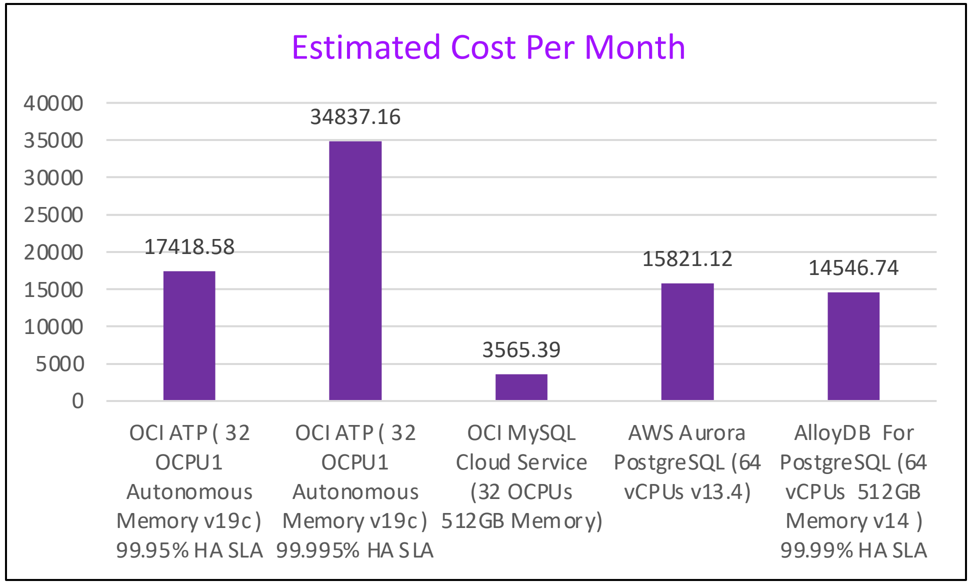


Figure 4 – Cost per month by database provider

Database	vCPU	OCPU	Memory(GB)	HA	Storage (TB)	License Included	Shared/Dedicated	Peak OCPU	Peak OCPU Duration % of mo	Baseline IO/sec	Peak IO/sec	Duration of peak hours/month	Total
OCI ATP (32 OCPU1 Autonomous Memory v19c) – 99.95% HA SLA	-	32	512	Y	1	Y	Shared	32	30	-	-	-	17418.58
OCI ATP (32 OCPU1 Autonomous Memory v19c) – 99.995% HA SLA	-	32	512	Y	1	Y	Shared	32	30	-	-	-	34837.16 + Data Guard
OCI MySQL Cloud Service (32 OCPUs 512GB Memory)	-	32	512	Y	1	-	-	-	-	-	-	-	3565.39
AWS Aurora PostgreSQL (64 vCPUs v13.4)	64	-	512	Y	1	-	-	-	-	4774	11935	210	15821.12
AlloyDB For PostgreSQL (64 vCPUs 512GB Memory v14) – 99.99% HA SLA	64	-	512	Y	1	-	-	-	-	-	-	-	14546.74

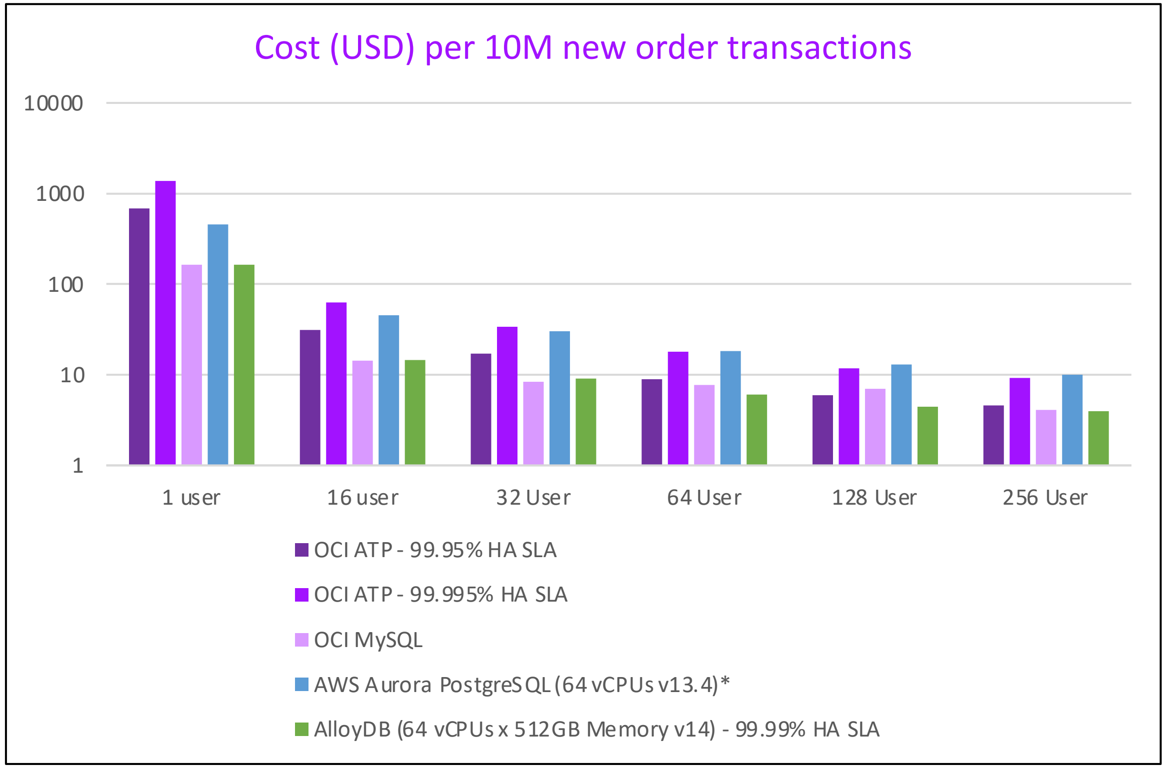


Figure 5

**AWS Aurora charges for IOs. At 256 users there were ~786,930 transactions. We assumed 91% of the total transactions were writes based on transaction types and the remaining 9% were reads of which 30% were not cached. We assumed 30% of the time was at peak and the baseline number is 40% of peak. Additional details in Figure 4 and the Appendix*

Using the output of the benchmarking results and the cost of each database as published in October of 2022, the cost per 10,000,000 transactions was calculated across platforms. To do this we first calculated the monthly cost of each database (see Appendix for Pricing configurations) assuming high availability configurations with the respective pricing tools and then divided by 43800 to define the per-minute cost:

	Cost per month	Cost per minute
Autonomous Database / DBCS (RAC) – 1TB Storage – 99.95% HA SLA	17419	0.40
Autonomous Database / DBCS (RAC) – 1TB Storage – 99.995% HA SLA	34837.16	0.80
OCI MySQL – 1TB Storage	3565	0.08
AWS Aurora PostgreSQL db.r6g.16xlarge +1TB Storage*	15821	0.36
AlloyDB 64 vCPU - 512GB Memory - 1TB storage – 99.99% HA SLA	14547	0.33

From here we calculated the cost per 10M new order transactions utilizing the data captured from the benchmarking results for the higher performing databases ((10000000/NOPM*per-min cost)*) with the results depicted in Figure 4.

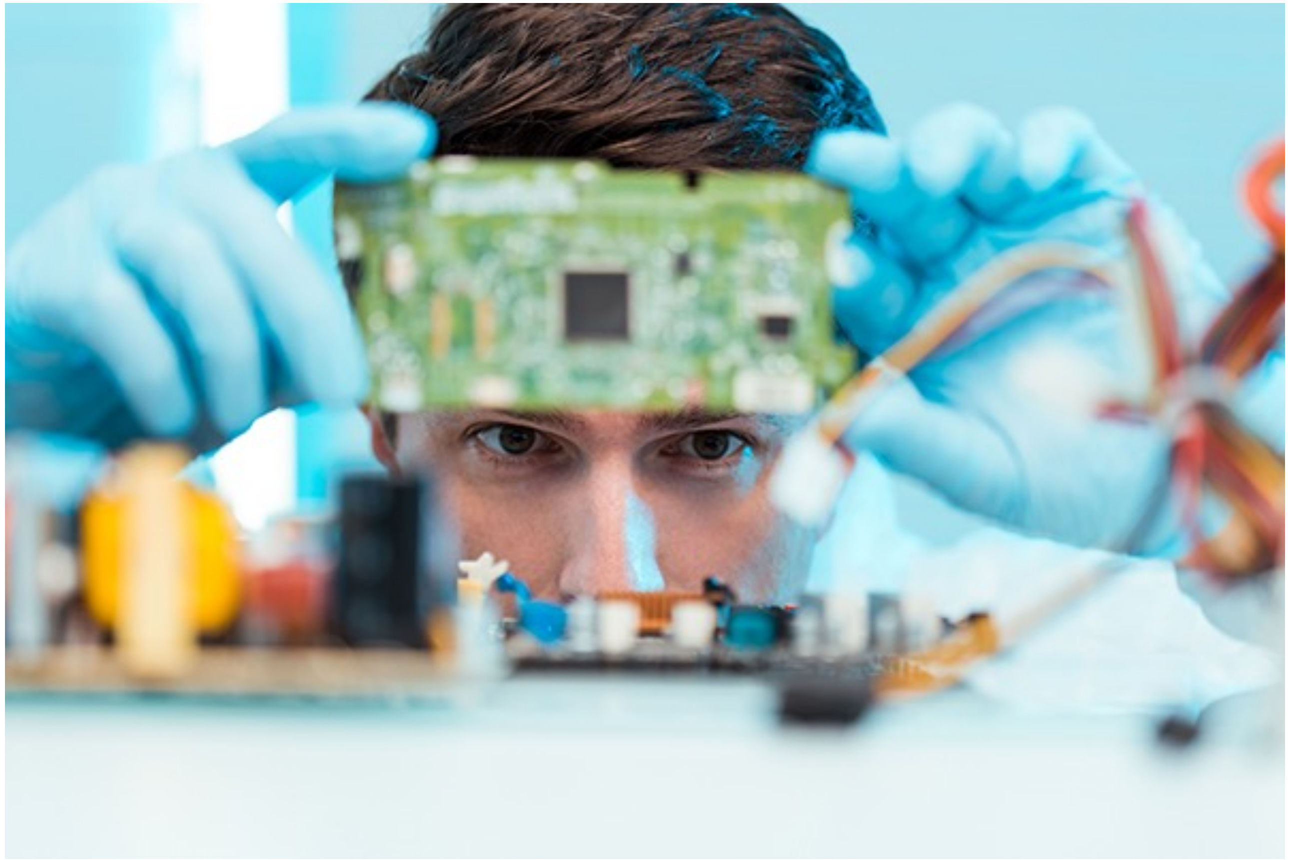
Running analytics and coexistence with transactional capacity without transactional performance impact

Columnar Engine

PostgreSQL has rich querying capability in addition to is transaction processing and for this reason PostgreSQL is also used for analytics. The challenge with using a traditional RDBMS for both transactional and operational use cases, is the tuning of the database. Additional indexes needed to support analytics can have an impact on transactional workloads. To solve for this, AlloyDB provides a columnar engine that can be enabled to support analytical queries while not impacting the transactional load.

AlloyDB utilizes machine learning to automatically organize data based on your workloads. Once the columnar engine is enabled, it utilizes machine learning to determine what data to keep in columnar format and learns as your workloads change. The engine can utilize the metadata it stores and the workloads being submitted to determine the best query plan for the request submitted.

This capability was tested and the results are below.



Setup

The columnar engine was enabled on the read pool instance after the initial benchmarking tests were run. After enabling the columnar engine, analytical queries were submitted against the database for approximately 30 minutes. The google_columnar_engine_run_recommendation was then executed to determine the recommended size for the columnar engine. The google_columnar_engine.memory_size_in_mb flag was set to the recommendation. Next the OLAP queries were run again and then the google_columnar_engine_run_recommendation was executed to recommend and populate the column store.

Results

Now that the columnar engine was configured, a benchmarking test using the configuration described above was run to get a baseline with the columnar engine enabled. A second run was executed and OLAP queries were also submitted against the database via a psql client and the results of the benchmarks with OLAP queries running against the tables were recorded. There was no significant change in benchmark performance on the primary. There was some lag on the replication to the read instance that resulted in the need to resubmit OLAP queries. This is a known issue and Google is working to address. Overall, we saw minimal impact on the OLTP load and combine this outcome with the built-in integration to Vertex AI, analytics and ML can be used to better serve your customers.

Performance Improvement after Enabling AlloyDB Columnar Engine

50x

Performance

```
select sum(ol_amount)
from order_line
where ol_i_id = 31627
```

34x

Performance

```
select count(*)
from order_line
where
ol_amount is not null ;
```

962x

Performance

```
select d_name,
    sum(ol_amount)
from district
inner join order_line
on (d_w_id = ol_w_id
      and d_id = ol_d_id )
where ol_i_id = 31627
group by d_name
order by 2 desc
limit 5;
```

2x

Performance

```
select ol_w_id,
    sum(ol_amount)
from order_line
where ol_i_id = 31627
group by ol_w_id
limit 5;
```

Use Cases for AlloyDB

There are multiple use cases for AlloyDB.

Applications that require high performance and transactional consistency in a managed service including enterprise applications, SaaS applications, cloud native start-ups, migrations from standard PostgreSQL implementations, cloud SQL for PostgreSQL, and other CSPs, and legacy applications being modernized.

AlloyDB as the database for an implementation of a new enterprise application: If your company is starting on a journey with an enterprise application, then AlloyDB can be an option to evaluate. Performance, cost, and in-database analytics are all items to consider when selecting the right database for your needs. AlloyDB matches or exceeds Oracle ATP at <256 users for a lower cost.

Modernizing an existing enterprise application that runs on a standard PostgreSQL database or MySQL on-prem or in the cloud: With little to no application code to update, focus can be on the key features offered by AlloyDB including high availability and automated administrative tasks. These features can be leveraged to free-up your human capital to work on new features in the system or other mission critical tasks.

Reimagining an existing legacy environment: If you have multiple, siloed legacy information systems used to run your business and are looking to modernize into a single application, AlloyDB should be considered an option for you application modernization project.

Summary

AlloyDB performs at >800k new order transactions per minute at a lower cost than other cloud options making it suitable for multiple use cases.

Combined with the automated administration tasks, high availability and scalability, and in-database analytics AlloyDB is an option in this space. When you combine the AlloyDB performance and cost with the other features you get from Google Cloud including:

- Extreme Network performance (private fiber connection between datacenters)
- Fully managed Kubernetes service for managing your containerized applications
- State-of-the-art security
- Vertex AI integration
- Sustainability (Google has been carbon neutral since 2007 with the goal to run o carbon free energy by 2030)

AlloyDB is an Enterprise ready database for new applications, modernizing existing applications, or refactoring legacy data silos into a centralized point.

References

- Google. (2022, 08 17). *Google Cloud AlloyDB*. Retrieved from Google Cloud: https://cloud.google.com/alloydb
- HammerDB. (2022, 08 16). *What is the TPC and then TPROC-C workload derived from TPC-C?* Retrieved from HammerDB Web Site: https://www.hammerdb.com/docs/ch03s02.html