FUTURUM.

Oracle Deploys MySQL HeatWave on Amazon and Scorches AWS on its Own Cloud

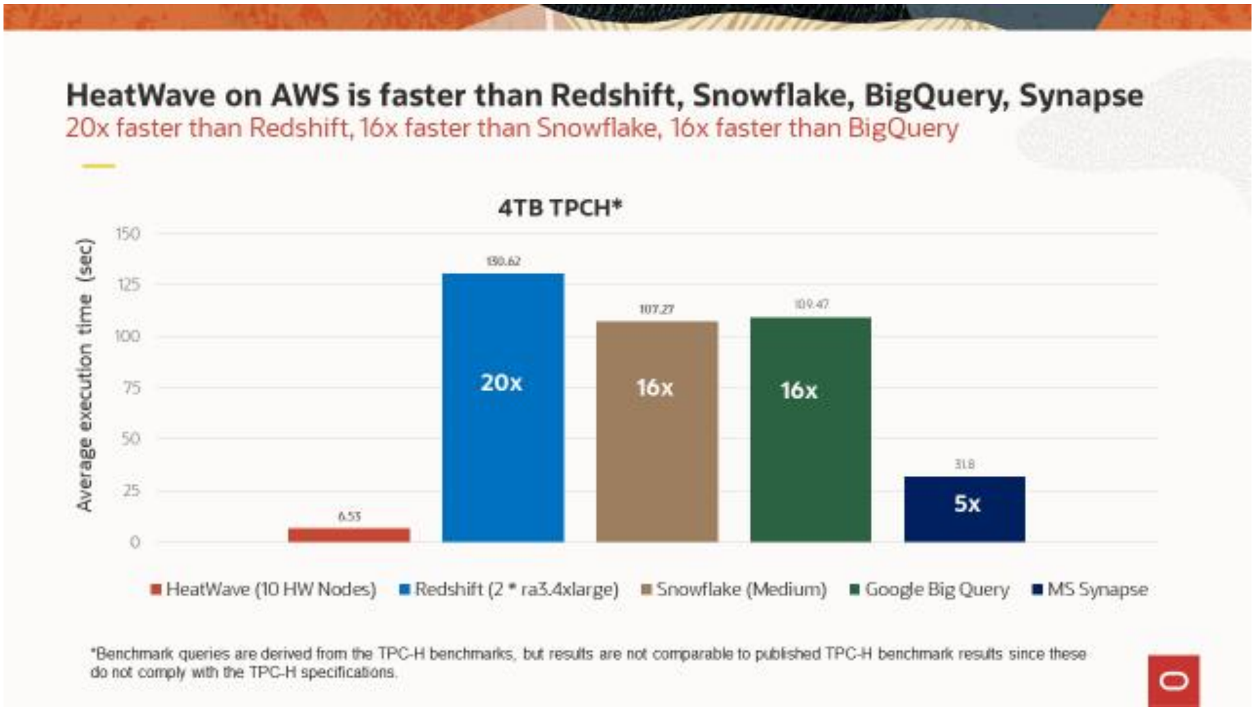
The News: Oracle announced that MySWL HeatWave is available on Amazon Web Services (AWS). MySQL HeatWave is designed to combine OLTP, analytics, machine learning, and machine learning-based automation within a single MySQL database. AWS users can now run transaction processing, analytics, and machine learning workloads in one service, without requiring time-consuming ETL duplication between separate databases such as Amazon Aurora for transaction processing and Amazon Redshift or Snowflake on AWS for analytics and SageMaker for machine learning. Read the Oracle press release here.

Oracle Deploys MySQL HeatWave on Amazon and Scorches AWS on its Own Cloud

Analyst Take: Oracle MySQL HeatWave on AWS debuts with a sales and marketing bang by offering customers a single database for transaction processing, analytics, and machine learning on the AWS hyperscale cloud platform, providing real-time analytics while eliminating ETL inefficiencies and costs. The immediate benefits also include full management capabilities that automate common tasks such as provisioning, patching, upgrades, and backups. The service runs natively on AWS — including the data plane, control plane and the console, all run in AWS providing AWS users with a true native experience.

In addition, HeatWave ML automates the machine learning lifecycle — building, training, tuning, and explaining ML models within the database. MySQL Autopilot provides the ML-based automation required to deliver auto provisioning, auto parallel load, auto data placement, and auto shape prediction capabilities aimed at improving both performance and user experience. From my view, all these benefits of the single database MySQL HeatWave on AWS and demonstrated price performance results can help accelerate adoption among AWS users of Aurora and Redshift as well as existing Oracle customers who use AWS for their cloud database needs.

As I assess qualitative and quantitative comparisons, note that I am basing my assessment on data presently made available to us. As a result, I see MySQL HeatWave on AWS delivering the speed and value advantages that customers prioritize. For instance, based on fully transparent and publicly accessible TPC-H benchmarks on GitHub, HeatWave is 20x faster than Amazon Redshift, 16x faster than Snowflake, and 16x faster than BigQuery, and 5x faster than Synapse in average execution time:



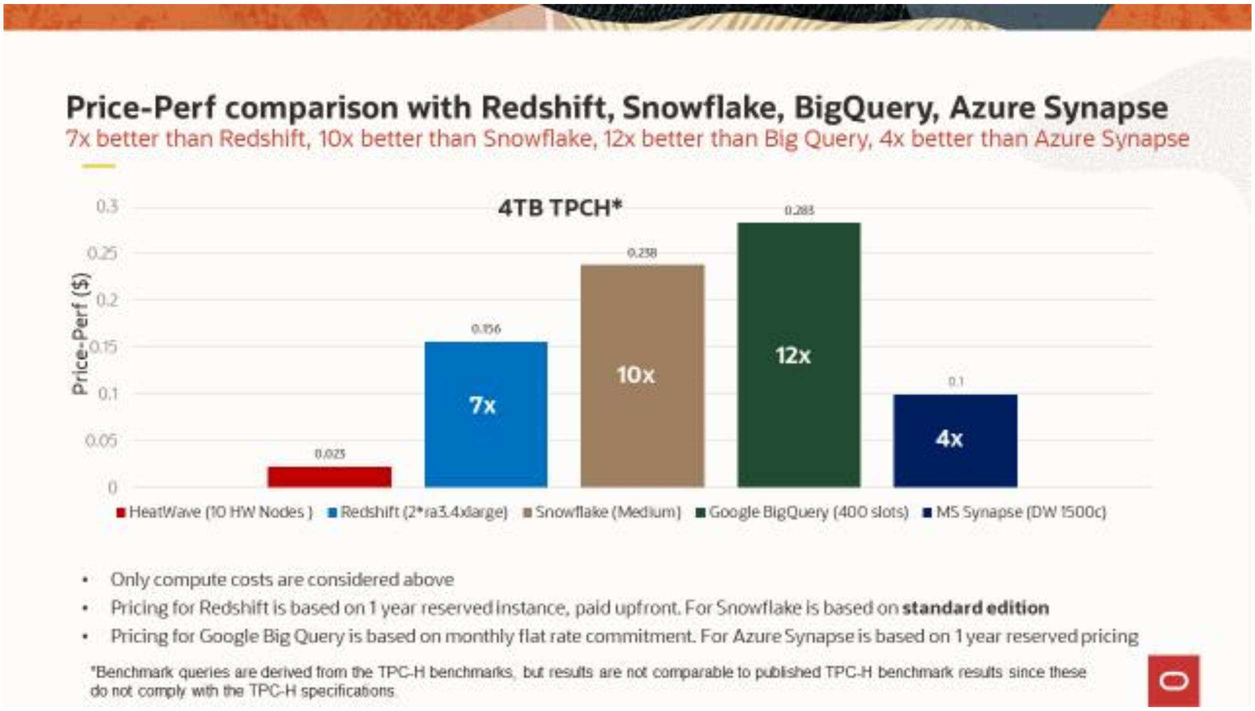
HeatWave on AWS is faster than Redshift, Snowflake, BigQuery, Synapse

20x faster than Redshift, 16x faster than Snowflake, 16x faster than BigQuery

*Benchmark queries are derived from the TPC-H benchmarks, but results are not comparable to published TPC-H benchmark results since these do not comply with the TPC-H specifications.

Source: Oracle

Likewise, based on compute costs, HeatWave on AWS delivers price-performance advantages such as 7x better than Redshift, 20x better than Snowflake, 12x better than BigQuery, and 4x better than Azure Synapse:



Price-Perf comparison with Redshift, Snowflake, BigQuery, Azure Synapse

7x better than Redshift, 10x better than Snowflake, 12x better than Big Query, 4x better than Azure Synapse

- Only compute costs are considered above
- Pricing for Redshift is based on 1 year reserved instance, paid upfront. For Snowflake is based on **standard edition**
- Pricing for Google Big Query is based on monthly flat rate commitment. For Azure Synapse is based on 1 year reserved pricing

*Benchmark queries are derived from the TPC-H benchmarks, but results are not comparable to published TPC-H benchmark results since these do not comply with the TPC-H specifications

Source: Oracle

What all this translates to is the fact that Oracle took its MySQL HeatWave cloud database software that was running on its infrastructure in OCI and architected an environment on AWS to run the same software and ended up outperforming Amazon's own top tier cloud databases in the process — including both Redshift and Aurora. This is the functional equivalent of a Lexus LF-A outperforming a Porsche 911 GT3 on its own test track in Weissach, Germany — you're not supposed to lose in your own backyard on the track you built from scratch.

MySQL HeatWave on AWS: HeatWave ML Key Differentiator

Of top importance, the in-database machine learning support in MySQL HeatWave is 25x faster than Redshift ML. This lets users train their model faster, which means that model can be retrained and kept up-to-date with data changing frequently. Having models which are more up-to-date results in improved prediction accuracy. I believe it is important to understand that HeatWave ML natively supports AutoML, including integration and completion criteria capabilities, and is convergence-based. In contrast, Redshift ML requires users to use Amazon SageMaker with the meter running, as its use is based on time budgets. Snowflake ML does not support AutoML at all, requiring users to rely on third-party libraries or chargeable services.

Both Redshift ML and Snowflake ML require users to move both data and models outside of the database, which increases security risks by exposing more surface area attack possibilities and makes data stale by the time all the data movement and manipulation is completed *before* data can be analyzed. Consideration must also be given to costs incurred when using separate ML tools and the learning curves associated therewith, which can result in lost employee productivity when evaluating potential solutions.

Moreover, MySQL Autopilot capabilities secure machine learning-based automation for MySQL HeatWave on AWS workloads. This includes supporting the system setup, data load, failure handing, and query execution functions critical to providing data-driven, query-driven, and ML automation features.

MySQL HeatWave on AWS: Automating Greater OLTP Benefits

MySQL Autopilot provides workload aware admission control that optimizes online transaction processing (OLTP) outcomes by reducing transaction contention. Auto Thread Pooling provides higher and sustained throughput at high concurrency by determining the optimal number of transactions which should be executed. As a result, MySQL HeatWave offers up to 10x better throughput than Aurora for OLTP. Through Auto Shape Prediction for OLTP, Autopilot continuously monitors the workload and recommends the right compute shape at any given time and determines the optimal shape which should be provisioned to provide the best price performance for OLTP workloads—continue using the existing shape, to upgrade to a larger shape for better performance or to downgrade to a smaller shape to reduce costs—whichever shape provides the best price performance.

From my perspective, the auto shape prediction capabilities deliver the price performance for OLTP advantages that fulfill topmost OLTP demands including buffer pool rate hits near 100%. I see this capability plus Auto Thread Pooling as linking directly to MySQL HeatWave on AWS offering up to 10x better throughput than Aurora for OLTP.

MySQL HeatWave on AWS: Security and Shape Considerations and Benefits

Of key importance, I view MySQL HeatWave on AWS as providing more security safeguards than Aurora due to topmost considerations such as support for immediate CPU security patches, while Aurora security patches can lag by an alarming six to nine months. In addition, MySQL HeatWave on AWS delivers enterprise-level masking, instance-based SQL firewall, strict administration privileges, asymmetric encryption, and multi-factor authentication vital to safeguarding and assuring cloud database security. These features are implemented inside the MySQL HeatWave server and not on top of the server, as is the case for some other databases.

Moreover, MySQL HeatWave on AWS provides support for a smaller HeatWave shape — i.e., 16GB shape for accelerating smaller workloads. I find customers want to use HeatWave to analyze small data sizes and that many OLTP databases already deployed are small. This smaller shape gives customers the flexibility to use the HeatWave node count that aligns with their actual workload requirements and avoid being shoehorned into much larger shape sizes. For customers, that's a big win.

Key Takeaways: Oracle MySQL HeatWave on AWS Red Carpet Debut

From my perspective, the MySQL HeatWave engineering team continues to execute at an accelerated pace. During their product development super cycle, they are making other cloud database service providers look rather complacent by comparison. While I expected this pattern on OCI — the company's home turf—the new capabilities available natively on the AWS platform take HeatWave to a whole new level. These include a rich interactive console that enables customers to run queries and administer and monitor the performance of resources. Within the console, they can also access MySQL Autopilot capabilities such as auto-provisioning. MySQL Autopilot, on its own, received numerous enhancements, including the ability to reduce resource contention and to generate auto shape predictions — all to accelerate OLTP performance.

Of course, I believe it would not be a HeatWave announcement without benchmarks demonstrating breakneck speeds and ultra-low costs. Oracle doesn’t disappoint in price performance, as MySQL HeatWave on AWS is 7x better than Amazon Redshift, 10x better than Snowflake, 12x better than GCP BigQuery, and 4x better than Azure Synapse. And in ML, it’s 25x faster than Redshift ML.

Finally, MySQL HeatWave on AWS already has a multitude of customer endorsements, including Johnny Bytes, 6D Technologies, ITSP, and Centroid, providing Oracle with a critical sales and marketing debut boost to help accelerate and broaden ecosystem adoption of the solution. Taken together, a robust HeatWave warning clearly remains in effect across the cloud database landscape.