db insight

RESEARCH NOTE
MySQL Heatwave engages Autopilot

Machine learning puts MySQL on overdrive

Executive Summary

Trigger

Last fall, Oracle introduced a new HeatWave compute engine that took cloud MySQL into new territory with analytics. For its next act, Oracle is applying machine learning to optimize the running of the database with a new Autopilot feature that will in some cases assists users and applies closed-loop automation to handle others. And with the second rev of the MySQL HeatWave service, available only in Oracle Cloud Infrastructure (OCI), Oracle is also more than doubling the upper capacity limits and improving linear scalability for query processing by 20%.

Our Take

For customers, Oracle's initial Heatwave release was designed to deliver OLAP *and* OLTP on a platform heretofore only known for transaction processing. For Oracle, HeatWave was intended to raise the company's profile in the MySQL landscape. Sure, Oracle has owned MySQL for well over a decade (as a result of its acquisition of Sun Microsystems) and keeps contributing code to the MySQL open source project. However, prior to HeatWave, Oracle never gave the MySQL installed base a compelling reason to move. HeatWave provided that reason – no other MySQL provider offers a credible solution for analytics.

The second rev of the HeatWave-powered Oracle MySQL Database Cloud Service, now roughly eight months old, solidifies Oracle's claim by adapting its expertise with ML to optimize a database, and it also roughly doubles the maximum cluster scale and memory size. Make no mistake about it, with double the scale and ML-based optimization, Oracle MySQL HeatWave is very much about reducing cost, increasing simplicity, and introducing analytics to organizations that already have MySQL and are looking for solutions to run them more economically, more simply, while adding real-time analytics. Oracle also plans for HeatWave to compete with other open source databases based on cost and operational simplicity.

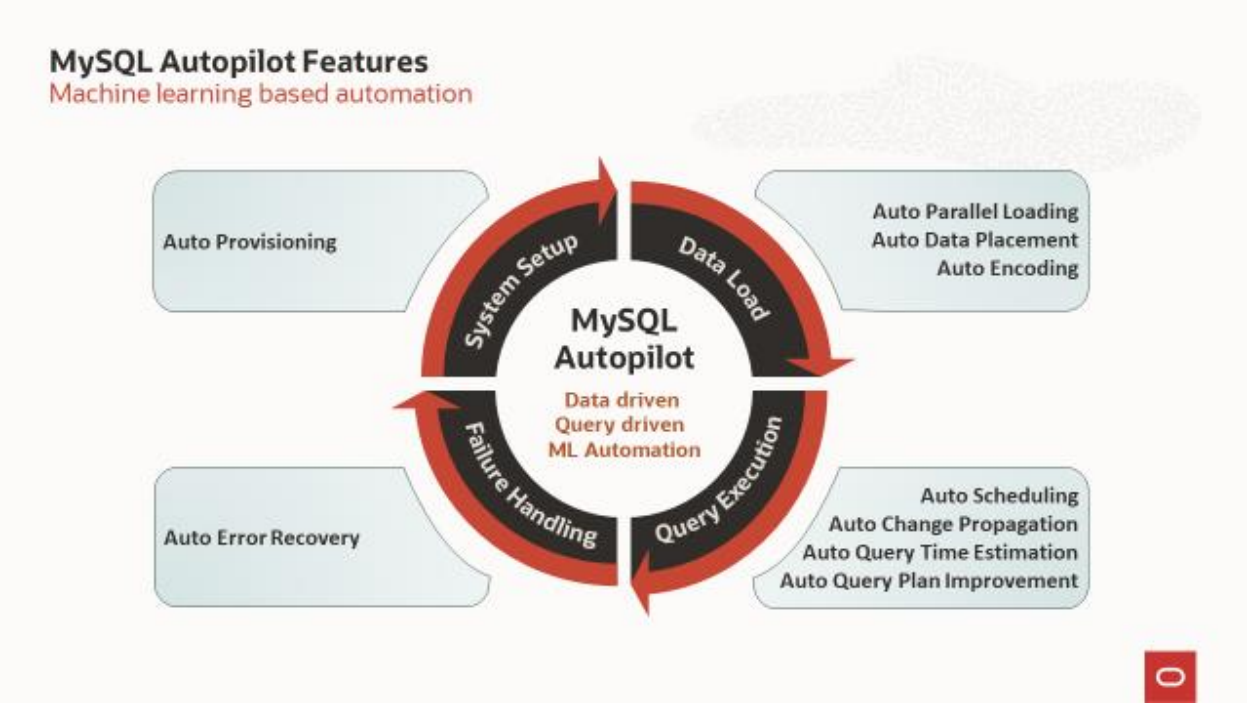
Incorporating Machine Learning under the hood

We have always believed that database *operations* provide ideal targets for applying machine learning, especially in the cloud. To be clear, this is quite different from adding the ability to run machine learning models inside the database. There are clear use cases for both, but for reasons that we will outline below, we believe that the use of ML internally presents the biggest bang for the buck for MySQL customers.

Internal operational optimization is the path that Oracle took for its second rev of the MySQL HeatWave service that it introduced last December. It was a natural next step for Oracle given the fact that its goal with HeatWave was not to introduce just another MySQL service. Having traversed into analytics, optimizing performance and cost of ownership are logical paths in a market where – Amazon Aurora for MySQL aside – most of the differentiation with third-party cloud MySQL cloud services is not with the core database engine, but instead is with implementation details such as super user access privileges, common consoles, replication approach, point-in-time recovery, and so on.

The new Autopilot feature that headlines the second release of MySQL HeatWave service introduces a mix of automation and smart recommendations targeting provisioning, error recovery, data loading, and query execution, as shown in Figure 1.

Figure 1. Oracle MySQL Autopilot features



Source: Oracle

More specifically, MySQL Autopilot factors in models addressing query performance, cluster capacity, network overhead, directory size, and load parallelism to perform the following tasks:

- **Auto provisioning –** The customer points the model to the tables that are likely to be queried the most frequently, and then Autopilot scans a small sample of data. The result is that the model predicts the amount of memory (and necessary cluster sizing) that will be needed based on the footprint of the table and presents the result as configuration recommendations that the customer can either accept or reject.
- **Auto data placement (loading) –** This feature *samples recent queries* to recommend where to physically place data and how to partition it. Specifically, it suggests placing rows and tables that are most frequently joined within the same (or adjoining partitions) of the same node, and pinpoints which columns should be partitioned in memory. Based on the assumption that query patterns can vary, this feature can kick in during runtime to help the customer make the requisite midcourse adjustments on-the-fly.
- **Auto query plan improvement –** Based on queries *that have already run* along with information on parameters such as average sizes of columns and rows, MySQL Autopilot iteratively improves query plans in a far more flexible way vs. traditional rules-based approaches, because with ML, it will literally learn from each new pass. These refinements are conducted automatically without requiring intervention from the user.
- **Auto scheduling –** Because HeatWave introduces analytic workloads to a database long known for transaction processing, queries to MySQL will become more varied. Many will be the familiar short, interactive queries associated with OLTP databases, while others will be more complex (and often, long-running) associated with analytic databases, such as quarterly reports. The goal with auto scheduling is to avoid holding up short, interactive queries because a complex query is soaking up the resources, and because users are likely to be more sensitive to latency issues with shorter queries. Autopilot predicts the execution time of each query, then gives short-running queries scheduling priority over longer ones. This feature is also automatic, not requiring end user intervention.

How ML boosts performance and reduces TCO

The notion behind applying machine learning is that it can scale the insights that would otherwise come with user experience over time and be performed manually, and as a result, more effectively optimizes the data to boost performance and lower its ownership costs.

So, while Oracle has been a longtime leader in enterprise databases on-premises, it has played the role of challenger on the cloud side. It's not surprising that in cloud, Oracle has priced aggressively ever since it introduced the Autonomous Database, where it has issued contractual guarantees to undercut Amazon Redshift. The headline is Oracle's claims that MySQL HeatWave, with Autopilot, can improve TPC-H performance up to 40%.

While Oracle currently isn't putting pricing guarantees in writing for MySQL HeatWave, it is hauling out the independent third party benchmarks. Our take on benchmarks is they are moving targets and therefore not set in concrete. Oracle is adopting what is becoming common and expected industry practice by publishing all of the benchmark scripts that were used; making them publicly available on GitHub; and using configurations published by each of the rival databases. It lets customers try these scripts themselves if they won't take the vendor's word for it.

For this go round, Oracle commissioned a third party to run the numbers. Compared with Amazon Aurora for MySQL, they showed Oracle delivering faster performance for 100GB *mixed* workloads and equivalent performance for 100GB *transaction processing* workloads, with both delivered at less than half the cost. For analytics, Oracle compared itself to Amazon Redshift, Snowflake, Azure Synapse Analytics, and Google BigQuery. Looking at the Redshift and Snowflake numbers, Oracle claimed up to 13x price/performance over Redshift AQUA (which includes memory optimization) and 35x better price/performance compared to Snowflake. These tests were conducted with 10-TByte data sets. By comparison, at the end of 2020, AWS ran its own benchmarks on Redshift (also published on GitHub) with smaller 3-TByte data sets, claiming up to 3x better price/performance to other unnamed analytic databases on the TPC-DS benchmark.

Adding scale

Because the cloud makes it easy to scale, in the form of inexpensive infrastructure and, thanks to improvements in underlying data engines to take advantage of linear scalability, the typical path for most cloud database platforms is to take advantage of that scale. In the second rev of MySQL HeatWave, Oracle has more than doubled maximum scale, clusters up to 64 nodes and data size processed to 32 TBytes. There are also tweaks to processing, which delivered a 20% improvement in linear scaling over rev 1.

With rev2, Oracle is also adding support for underlying storage approaches supporting scale. Columnar data not stored in-memory is kept in the same format in OCI object storage. That allows bi-directional data traffic between memory and storage to be highly granular and performant, and if there are any outages, it allows loading to pick up where it left off, enabling fast recovery, with data reloading into HeatWave by up to 100 times faster than the prior release.

Takeaways

Oracle has taken logical next steps with the MySQL HeatWave service by introducing under-the-hood machine learning, which doubles down on its value proposition. Last fall, Oracle demonstrated that MySQL could become a mixed workload database service without excuses. Introducing machine learning under the hood, for which Oracle has experience, is a logical means for reducing TCO – which is a key criterion for MySQL users. It makes sense that Autopilot is intended to optimize performance, not automate the running of MySQL, because there is not a crying need to make smaller-scale databases self-driving.

With Autopilot, Oracle is eating its own dogfood by applying ML to optimize database operation to deliver simplicity and lower cost of ownership. Oracle first introduced ML into database operation with the Autonomous Database (ADB). HeatWave is a second act, but a very different one, targeted at different database footprints (that max out at roughly 10x smaller than ADB); uses cases (departmental applications instead of enterprise applications like Oracle Fusion ERP); and approaches for applying ML (providing smart assistance as opposed to self-driving databases).

This is still early days for Oracle MySQL HeatWave. For instance, rivals such as Azure Synapse Analytics and Snowflake support the capability to run Spark workloads natively, while Amazon Redshift and Google BigQuery can now run machine learning models in-database (as noted above, this is different from using machine learning to run or optimize operation of the database). Extended analytics should be next on Oracle's to-do list for HeatWave.

With the Autonomous Database, Oracle has proven that it can draw in non-Oracle customers to Oracle. The customer references shown at the MySQL HeatWave rev 2 announcement demonstrate that Oracle can also pull in non-Oracle customers, such as Amazon Aurora users. However, we believe that a much larger sweet spot for MySQL HeatWave will be Oracle's existing customer base. In all likelihood, they have dozens if not hundreds of MySQL databases floating around their organizations and they've never heard about HeatWave, yet.

Author

Tony Baer, Principal, dbInsight

tony@dbinsight.io

Twitter @TonyBaer

About dbInsight

dbInsight LLC® provides an independent view on the database and analytics technology ecosystem. dbInsight publishes independent research, and from our research, distills insights to help data and analytics technology providers understand their competitive positioning and sharpen their message.

Tony Baer, the founder and principal of dbInsight, is a recognized industry expert on data-driven transformation. *Onalytica* named him as one of its influencers for data, data management, and cloud in 2019, 2020, and 2021. *Analytics Insight* named him one of the 2019 Top 100 Artificial Intelligence and Big Data Influencers. His combined expertise in both legacy database technologies and emerging cloud and analytics technologies shapes how technology providers go to market in an industry undergoing significant transformation. His regular ZDnet *"Big on Data"* posts are read 25,000 – 30,000 times monthly.