MySQL HeatWave on AWS

One MySQL Database for OLTP, OLAP, and Machine Learning on AWS

September 2023
Copyright © 2023, Oracle and/or its affiliates
Purpose statement

This document provides an overview of features and enhancements included in HeatWave. It is intended solely to help you assess the benefits of HeatWave and to plan your I.T. projects.

Disclaimer

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle. Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Benchmark queries are derived from the TPC-H and TPC-DS benchmark, but results are not comparable to published TPC-H and TPC-DS benchmark results since they do not comply with the TPC-H TPC-DS specification.
Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purpose statement</td>
<td>2</td>
</tr>
<tr>
<td>Disclaimer</td>
<td>2</td>
</tr>
<tr>
<td>Introduction</td>
<td>4</td>
</tr>
<tr>
<td>Performance and Price Performance Advantages</td>
<td>5</td>
</tr>
<tr>
<td>Architecture of MySQL HeatWave on AWS</td>
<td>6</td>
</tr>
<tr>
<td>MySQL HeatWave Control Plane</td>
<td>6</td>
</tr>
<tr>
<td>MySQL HeatWave Data Plane</td>
<td>7</td>
</tr>
<tr>
<td>HeatWave Scale-Out Data Management</td>
<td>7</td>
</tr>
<tr>
<td>Integration with AWS Services</td>
<td>8</td>
</tr>
<tr>
<td>MySQL HeatWave Features on AWS</td>
<td>8</td>
</tr>
<tr>
<td>MySQL Database</td>
<td>9</td>
</tr>
<tr>
<td>HeatWave Analytical Processing</td>
<td>13</td>
</tr>
<tr>
<td>HeatWave Lakehouse</td>
<td>13</td>
</tr>
<tr>
<td>HeatWave AutoML</td>
<td>14</td>
</tr>
<tr>
<td>Current challenges of Machine Learning in databases</td>
<td>14</td>
</tr>
<tr>
<td>HeatWave AutoML Approach</td>
<td>15</td>
</tr>
<tr>
<td>HeatWave AutoML Advantages</td>
<td>15</td>
</tr>
<tr>
<td>MySQL HeatWave Console</td>
<td>16</td>
</tr>
<tr>
<td>Lifecycle Management</td>
<td>16</td>
</tr>
<tr>
<td>Interactive Schema and Data Management</td>
<td>17</td>
</tr>
<tr>
<td>Interactive Query Interface &amp; Workload Monitoring</td>
<td>20</td>
</tr>
<tr>
<td>MySQL Autopilot Auto Shape Prediction Advisor</td>
<td>21</td>
</tr>
<tr>
<td>Autopilot Indexing</td>
<td>22</td>
</tr>
<tr>
<td>Performance Monitoring</td>
<td>23</td>
</tr>
<tr>
<td>HeatWave AutoML Model Creation, Prediction, and Explanation</td>
<td>24</td>
</tr>
<tr>
<td>Conclusion</td>
<td>24</td>
</tr>
</tbody>
</table>
Introduction

MySQL HeatWave on Amazon Web Services (AWS) is a fully managed database service. The service automates the common management tasks of a database system, such as:

- Setting up, configuring, and tuning the database
- Security patching and upgrading the host operating system and database system
- Orchestrating database backups

MySQL HeatWave is the only service that combines transaction processing, real-time analytics, and machine learning across data warehousing and data lake within one single database.

MySQL HeatWave eliminates the need for complex, time-consuming ETL operations and unnecessary data duplication between separate databases and tools for OLTP, analytics, and machine learning use cases. Customers avoid the latency and security risks of data movement between data stores while reducing costs. MySQL HeatWave also includes MySQL Autopilot, providing workload-aware, machine learning-powered automation of various aspects of the database system lifecycle; including provisioning, data management, query execution, and failure handling for both OLTP and analytics workloads.

Oracle makes all these MySQL HeatWave capabilities, which are built, managed, and continuously supported by the MySQL HeatWave development team, available on AWS. All components of the MySQL HeatWave service on AWS, namely the service console, control plane, and data plane, are built and optimized for AWS. MySQL HeatWave’s native integration with AWS enables customers with applications already deployed in AWS to benefit from MySQL HeatWave without incurring the latency associated with accessing a database service running outside of AWS. Customers also don’t incur the high data egress fees charged by AWS that would be necessary to migrate data to a service running outside of AWS. Lastly, the tight integration of MySQL HeatWave with AWS services such as Amazon S3, CloudWatch, and PrivateLink, makes it easy for developers to rely on MySQL HeatWave for new applications.

“MySQL HeatWave on AWS fits perfectly into our data platform with 60X to 90X faster complex queries compared to AWS RDS and Aurora. It generates real-time analytics we need for targeted, multichannel campaigns. We now have greater scalability to onboard more data and new clients of any size without increasing IT admin.”

Thomas Henz
Chief Executive Officer, Johnny Bytes
Performance and Price Performance Advantages

With its superior data-processing architecture and optimization for AWS infrastructure, MySQL HeatWave on AWS delivers unmatched performance and price-performance. On the 4TB TPC-H benchmark, MySQL HeatWave on AWS delivers 7x better price performance than Amazon Redshift, 10x better than Snowflake, 12x better than Google BigQuery, and 4x better than Azure Synapse. For machine learning, MySQL HeatWave on AWS is 25X faster than Redshift ML. On a 10GB TPC-C workload, MySQL HeatWave offers up to 10X higher and sustained throughput compared to Amazon Aurora at high concurrency. All of these fully transparent benchmark scripts are available on GitHub for customers to replicate.

“MySQL HeatWave on AWS has 50X faster complex queries compared to AWS RDS that provide us real-time insights to accelerate application development and help us improve patients’ lives.”

Kyle Yang
Assistant Manager
Bionime
MySQL HeatWave on AWS delivers a native experience for AWS customers. The console, control plane, and data plane completely reside in AWS and are responsible for managing the MySQL HeatWave database resources in AWS. The control plane communicates with Oracle Cloud Infrastructure (OCI) Identity for account management, and with OCI metering & billing for monitoring and managing the usage and expenses associated with the customer’s account.

Once the user signs up for an OCI cloud account and registers their OCI account with MySQL HeatWave on AWS, the main interactions with the MySQL HeatWave service take place in AWS, through the service console hosted at cloud.mysql.com.

MySQL HeatWave Control Plane

The MySQL HeatWave control plane, which enables the management of MySQL HeatWave and maintains the necessary metadata, is built on publicly available AWS services. The control plane components are hosted in the Oracle AWS account dedicated to the MySQL HeatWave service and are tightly controlled with AWS Identity and Access Management permissions and policies.

The MySQL HeatWave control plane is responsible for the management of the database system lifecycle including provisioning/de-provisioning/pausing, the configuration of the database system, orchestrating backups, security patching, upgrades, monitoring, as well as ensuring the isolation of different database systems from each other.
MySQL HeatWave Data Plane

MySQL HeatWave on AWS hosts all the customer databases components in a dedicated AWS account and strictly isolates them from the service control plane components and other database systems managed by the control plane.

MySQL HeatWave databases are hosted on AWS services, which are publicly available to AWS customers, such as Elastic Compute Cloud (EC2) for data processing, Elastic Block Storage (EBS) for storage, and VPC for resource isolation. Customers do not have access to the infrastructure on which the database runs, such as the database host machines or storage; they are provided with a MySQL endpoint, which can be accessed by a standard MySQL client. Customers can restrict the client addresses that can access their database systems through the MySQL endpoint and the connection between the client and database system is secured with TLSv1.2. Using the MySQL endpoint, customers can run any application against their MySQL HeatWave database systems.

HeatWave Scale-Out Data Management

MySQL HeatWave on AWS now provides an optimized storage layer built on Amazon S3 to store the HeatWave in-memory hybrid columnar representation of the data. This allows data to be reloaded to each HeatWave node independently and in parallel, significantly improving the service uptime and the performance of operations such as error recovery, maintenance, and system restart.

The fast and automatic data reload from HeatWave storage layer also enables customers to easily and quickly pause and resume a HeatWave cluster to save cost when HeatWave is not needed.
Integration with AWS Services

MySQL HeatWave on AWS integrates natively with various AWS services to provide a seamless integration with AWS applications.

AWS Identity and Access Management (IAM)

MySQL HeatWave on AWS integrates with AWS Identity and Access Management which enables MySQL HeatWave instances to securely access data or resources in customer’s AWS account. It uses the cross-account roles mechanism provided by AWS IAM service. With this mechanism, customers can delegate S3 access permission to a MySQL HeatWave instance, providing tighter security which is better suited for enterprise production workloads.

Amazon S3

Integration with Amazon S3 enables MySQL HeatWave instances to read or write data to S3 directly. It supports multiple use cases:

1. HeatWave scale-out data management - HeatWave in-memory representation of the data is stored to Amazon S3, enabling fast reload of data for restart, maintenance, pause and resume, and error recovery. As a result, this improves service uptime.

2. HeatWave Lakehouse – The integration with Amazon S3 enables HeatWave Lakehouse to securely access customer’s data files stored on Amazon S3, providing fast analytics query processing and machine learning capability directly on data stored on Amazon S3.

3. Data import from Amazon S3 – This enables users to directly import data stored on Amazon S3 to MySQL.

MySQL HeatWave Features on AWS

MySQL HeatWave provides a single database system for high-performance and secure transaction processing, real-time analytics, and machine learning. Each MySQL HeatWave database system consists of a MySQL Database node, mainly targeting transaction processing, and a configurable number of HeatWave nodes for analytical processing and machine learning tasks. MySQL HeatWave on AWS strives to provide the best price/performance by optimizing all components of the underlying AWS infrastructure—while keeping costs as low as possible.

“For cost conscious IT teams and developers, MySQL HeatWave on AWS represents a whole new TCO calculation with zero cost for what are add-on services on AWS and no data egress fees.”

Marc Staimer
Senior Analyst
Wikibon
MySQL Autopilot provides workload-aware, machine learning-based automation of various aspects of the application lifecycle, including provisioning, data management, query execution, and failure handling. Combined, these features improve the performance of the application, reduce cost by predicting the optimal configuration to run a workload, and reduce manual database administration. MySQL Autopilot’s machine learning models, hosted in the database system, leverage the run time information like data and query statistics, memory, and network usage, providing continuous improvements over time. In addition to the automation features offered for the HeatWave analytics engine, MySQL Autopilot includes capabilities designed for OLTP workloads, which further improve the MySQL HeatWave price-performance (further information provided below).

Oracle provides the latest MySQL Enterprise Edition version in MySQL HeatWave on AWS, which contains the state-of-the-art performance, functionalities, and security features developed, maintained, and supported by the MySQL team at Oracle. With each new release of the MySQL server, customers of MySQL HeatWave on AWS will have immediate access to the new features that are developed and tailored for the AWS infrastructure to maximize performance and security. Some of the performance and security features of MySQL HeatWave on AWS are presented in the following sections.

**MySQL Database**

**Tuned for peak performance**
MySQL HeatWave on AWS provides several shapes (i.e., AWS EC2 instance types) with different vCPU counts and memory sizes, on which the MySQL server can run for transaction processing. Customers can select a shape based on their workload needs. Each MySQL shape comes with a default MySQL configuration tailored for that shape to maximize performance on the target AWS infrastructure. Depending on the selected shape and storage size specified by the customer, the underlying storage system built on AWS EBS is tuned for peak performance without magnifying the associated costs.

**MySQL Autopilot features for OLTP**
MySQL Autopilot is now enhanced with two new features to provide workload-aware, machine learning-powered automation capabilities for OLTP workloads—to further improve performance and throughput while enabling cost savings:

“For any developers working with MySQL on AWS, Oracle has just dropped a big productivity boost on your doorstep without the big price tag.”

Carl Olofson
Research Vice President
IDC
1. **Autopilot Indexing**: MySQL Autopilot Indexing (currently in limited availability) recommends the right set of indexes for columns in order to improve OLTP query performance. It balances cost, storage space, and performance by adding or removing indexes.

With Autopilot Indexing, database administrators no longer need to manually identify which indexes are most beneficial for their workload. Autopilot Indexing automatically generates secondary index recommendations to create or drop indexes based on the current workload. Autopilot Indexing considers both the query performance and the cost of maintaining the indexes when generating recommendations. It provides performance and storage estimations, as well as explanations for the recommendations it generates. The Autopilot Indexing interface consists of a simple and intuitive console that customers can use to view and analyze the projected performance and storage impact of recommended index suggestions. This makes it easy to foresee the impact of changes to the database systems before applying the suggestions.

For more information, read the MySQL Autopilot Indexing Technical Brief.

2. **Auto Shape Prediction**: To alleviate the burden of experimenting with different MySQL shapes to determine the most performant shape for a given workload, Auto Shape Prediction provides suggestions for the right MySQL server shape, based on highly accurate predictions from machine-learning models inside the MySQL server, along with the most recent query execution metrics and traces. Since Auto Shape Prediction continuously collects workload execution statistics, it can adapt to the evolving workload patterns and provide suggestions based on the most recent workload.

3. **Auto Thread Pooling**: With Auto Thread Pooling, MySQL HeatWave prioritizes not only peak single-thread performance, but also high throughput in the presence of concurrent clients running concurrent queries on a MySQL server. With this feature, the MySQL server now can perform workload-aware admission control of the incoming transactions. It eliminates the resource contention created by too many awaiting transactions, automatically queuing them to maximize performance while sustaining the throughput in the face of high concurrency.

Advanced security and compliance features: MySQL HeatWave on AWS includes several comprehensive security features natively implemented in the MySQL server, as opposed to other services such as Amazon Aurora, which provide security methods as an additional layer on top of the database.
1. **Data masking and de-identification**: Helps organizations protect sensitive data from unauthorized users by hiding and replacing real values with substitutes.

2. **Asymmetric encryption**: Enables developers and DBAs to increase the protection of confidential data and comply with regulatory requirements including HIPAA, Sarbanes-Oxley, and the PCI Data Security Standard, through encryption, key generation, digital signatures, and other cryptographic features.

3. **Database firewall**: Provides real-time protection against database-specific attacks such as SQL injections, by monitoring, alerting, and blocking unauthorized database activity without any changes in the application.

**JavaScript Support (GraalVM):**
This feature (in limited availability) supports JavaScript stored programs in MySQL Heatwave. This enables developers to develop rich procedural logic inside the database and access their MySQL datasets seamlessly. The JavaScript stored programs are run in an environment running Oracle GraalVM Enterprise edition. GraalVM is an Oracle compiler eco-system that includes JDK, language implementation such as JavaScript, R, Python, Ruby, and Java. It includes just-in-time (JIT) and head-of-time (AOT) compilation technology and provides a fully managed virtual machine with sandboxing capability and tooling support.

**Data import from Amazon S3**
Data Import for MySQL Heatwave on AWS allows users to import data directly to MySQL Heatwave from Amazon S3. It provides a simple and intuitive user interface on the MySQL HeatWave AWS console, allowing users to easily import data from Amazon S3 to MySQL HeatWave. It supports data formats such as MySQL shell dump and delimited text files.

The data import capability enables improved ease-of-use, as well as enhanced security and better performance for importing data on Amazon S3 into MySQL HeatWave. It supports both AWS User Access Keys and AWS IAM Roles for authentication to Amazon S3. AWS IAM Roles grant exclusive access to user data on Amazon S3 to specific MySQL HeatWave instance(s), providing tighter security which is better suited for enterprise production workloads.
With this feature, users can more easily migrate data to MySQL HeatWave on AWS from MySQL-compatible databases such as AWS Aurora, AWS RDS, MySQL on AWS EC2 instance, or MySQL on-premises. Users also have the flexibility to export data from AWS Redshift, Snowflake, or Google BigQuery in delimited text files, and then import to MySQL HeatWave on AWS.

**Native Bulk Ingest of Data Files in Amazon S3 into MySQL HeatWave on AWS**
To provide a fast mechanism for importing large data sets from Amazon S3 to MySQL HeatWave, the MySQL native data import functionality has been enhanced to support direct import of data files stored on Amazon S3 into MySQL and a new Bulk Ingest algorithm to provide better data import performance with low memory usage.

As compared to AWS Aurora, importing a 1TB data set with data already sorted in the data file in Amazon S3, MySQL HeatWave on AWS is 12.4X faster. For data that is not sorted, MySQL HeatWave on AWS is 7.5X faster and uses 5.8X less memory than Aurora.

**Inbound Replication**
For organizations who want to keep their existing MySQL workloads (either on cloud, such as AWS RDS/Aurora, or on-prem) but want to use MySQL HeatWave for their dev/test, analytics and machine learning use cases, they can easily set up inbound replication to replicate their existing MySQL databases to MySQL HeatWave on AWS.

**AWS PrivateLink Support**

Replicating data from AWS RDS/Aurora and MySQL on-premise to MySQL HeatWave on AWS
With the support of AWS PrivateLink, organizations can connect their applications to MySQL HeatWave through a private connection. All traffic to and from MySQL HeatWave instance remains private, secured, and always stays within the AWS network. In addition, MySQL HeatWave on AWS uses AWS Identity and Access Management which enables customers to control creation of AWS PrivateLink VPC endpoints that are required to connect to MySQL HeatWave. This allows customers to have better control of their network configurations to MySQL HeatWave, as well as increased security.

**HeatWave Analytical Processing**

**Optimized and tuned for peak performance and best price performance**
As the MySQL server, the HeatWave analytical processing engine is tuned to achieve the best performance on the AWS infrastructure. To minimize costs on AWS, the HeatWave processing engine is enhanced with aggressive compression of the in-memory data, which helps to reduce the memory footprint, and hence the number of required processing nodes, without sacrificing peak performance.

**Support for small workloads**
HeatWave has been designed to meet the needs of very large datasets by utilizing many processing nodes with large memory capacities. To enable OLTP databases with smaller datasets to benefit from the capabilities of the HeatWave engine, MySQL HeatWave on AWS provides customers the option of selecting a cluster with a smaller HeatWave shape, enabling them to benefit from the high performance of HeatWave at a lower cost.

**HeatWave Lakehouse**

HeatWave Lakehouse allows efficient data query processing in Amazon S3 with data in the MySQL database. Users are not only able to query large amounts of data in Amazon S3 storage, they can also directly use AutoML capabilities in MySQL HeatWave on that data without leaving the comfort of HeatWave web console. Data in the object store remains in the object store and is not copied into
the MySQL database. It uses the same scale-out query processing engine of MySQL HeatWave, enabling it to query hundreds of terabytes of data in parallel from Amazon S3 in various file formats, including CSV, Parquet, Avro, or exports from databases like Aurora, Redshift, and even MySQL, without users having to import data into the database.

MySQL Heatwave Lakehouse benefits

1. Highly partitioned architecture to quickly process data from Amazon S3 without moving data out of AWS.
2. MySQL Autopilot support for automatically inferring schema from raw semi-structured data in CSV and Parquet formats.
3. Full suite of AutoML capabilities provided by HeatWave AutoML without data leaving MySQL HeatWave.
4. Secure access to customer data in Amazon S3 buckets using AWS IAM roles, where the customer has complete control over what data is shared with the MySQL Heatwave on AWS service.
5. Ability to provision large cluster sizes in matter of a few minutes.

To learn more about HeatWave Lakehouse, download the HeatWave Lakehouse Technical Brief.

HeatWave AutoML

Current challenges of Machine Learning in databases

Developing and using machine-learning models requires skill sets in topics such as:

- Candidate algorithms/models to select from
- Hyperparameters that need to be tuned per algorithm
- Features to engineer and select from
- Data preprocessing approach per data type
- Drift detection and retraining
- Knowledge of Python, as most ML algorithm frameworks are available only in Python
Even with the above expertise, users still need to extract data out of the database to train and test the model, which can lead to trust and security issues.

**HeatWave AutoML Approach**

HeatWave AutoML enables users to train ML models, generate inferences, and generate explanations across data stored in MySQL databases and data lakes. This is in-database machine learning, therefore training, inference, and explanation activities are performed inside the database, with no need to extract data out of the database. There are several advantages:

**HeatWave AutoML Advantages**

HeatWave AutoML enables MySQL users to train a model, generate inferences, and generate explanations, without extracting data out of the MySQL database. It provides several advantages:

1. **Fully Automated**: HeatWave AutoML fully automates the creation of tuned models, generating inferences, and generating explanations, thus eliminating the need for the user to be an expert ML developer.
2. **SQL interface**: Provides the familiar MySQL interface for invoking machine learning capabilities.
3. **Security and Efficiency**: Data and models never leave the MySQL Database. Clients or any other services never see the data or models stored in the DB service.
4. **Explanations**: All models created by HeatWave AutoML can be explained. Enterprises have a growing need to explain the predictions of machine learning models to build trust, demonstrate fairness, and comply with regulatory requirements.
5. **Performance and Scalability**: The performance of HeatWave AutoML is much better at a lower cost than competing services such as Redshift ML. Furthermore, HeatWave AutoML scales with the size of the cluster.
6. **Easy Upgrades**: HeatWave AutoML leverages state-of-the-art open-source Python ML packages that enable continual and swift uptake of newer (and improved) versions.
7. **Supported Models**: HeatWave AutoML supports multiple model types such as Classification, Regression, Time Series Forecasting, Anomaly
Detection, and Recommender System. This enables organizations to use HeatWave AutoML for different types of business use cases.

All these capabilities are available to MySQL HeatWave customers without any additional charge. MySQL HeatWave console on AWS enables HeatWave AutoML features through a user-friendly interface, further increasing the ease of use of the advanced HeatWave AutoML capabilities.

To learn more about HeatWave AutoML, download our technical brief.

**MySQL HeatWave Console**

The MySQL HeatWave console is designed to facilitate:

1. Lifecycle management of the MySQL HeatWave resources, such as MySQL Database systems, MySQL backups, MySQL configurations, and HeatWave clusters.
3. Workload monitoring through visualization of the schema metadata and statistics that are stored on the MySQL server.
4. Management and visualization of training, inference, and explanation of machine learning algorithms offered by HeatWave AutoML.

**Lifecycle Management**

Customers can manage the MySQL HeatWave resources associated with their accounts through the MySQL HeatWave console. The MySQL HeatWave console provides a single pane of glass to manage MySQL database systems, HeatWave clusters attached to the MySQL Database systems, configurations, backups and networking setup such as inbound replication channels and privatelinks of the MySQL database systems. The service console allows users to, for example, provision/deprovision/configure resources, pause a MySQL database system, and pause the HeatWave cluster attached to it if no user activity is expected to save costs.
Interactive Schema and Data Management

MySQL HeatWave on AWS can process data on MySQL or Amazon S3. The console provides intuitive user interfaces for users to load and manage data in MySQL HeatWave.

1. Manage data on Amazon S3 using HeatWave Lakehouse

   To process data on Amazon S3, HeatWave Lakehouse needs to first map the data to a MySQL table, as data is not stored in MySQL database system. HeatWave Lakehouse uses MySQL Autopilot auto schema inference to automatically infer the mapping of file data to data types in the database. As a result, customers do not need to manually specify the mapping for each new file to be queried by MySQL HeatWave Lakehouse, saving time and effort.

   ![Create Lakehouse mapping using the console](image1)

MySQL Autopilot – Auto Schema Inference suggests the schema with column name mapping, data type and precision

2. Import data on Amazon S3 to MySQL
Data import provides a simple way to import data to MySQL database systems such as MySQL dump or CSV files on Amazon S3 using the MySQL HeatWave on AWS console.

3. Replicate data from a source MySQL database using inbound replication

Customers can set up an inbound replication channel to replicate data from their existing MySQL deployment, such as AWS RDS/Aurora, to MySQL HeatWave. The console provides a simple interface to set up the replication channel, which internally uses MySQL native replication.

Once the data mapping is set up on Lakehouse or data is imported or replicated into the MySQL server, customers can leverage MySQL Autopilot’s Auto Provisioning capability to estimate the HeatWave cluster size needed for their workload.
To enable this, the MySQL HeatWave console provides users with a list of schemas and tables in the MySQL server. Depending on which tables the user wants to run analytical queries on, Auto Provisioning estimates the total memory usage in the HeatWave cluster memory, and based on the HeatWave shape selected by the user, estimates the number of HeatWave cluster nodes needed to accommodate the target dataset. Users can then provision a HeatWave cluster based on the number of nodes suggested by Auto Provisioning and load the desired tables to the HeatWave memory for analytical processing.

Customers can continuously monitor the schemas and tables loaded in the MySQL server as well as the HeatWave memory. This monitoring information helps loading the desired tables into the HeatWave cluster memory in the most efficient way. The user first needs to select which tables to load into the HeatWave memory and then use Auto Parallel Load, which optimizes the load time and memory usage of the data load operation into HeatWave by predicting the optimal degree of parallelism for the set of tables selected by the user. To provide better visibility into storage and memory usage, the MySQL HeatWave console also provides detailed information about the estimated memory footprint of each table in the HeatWave cluster memory, encoding type, load status, as well as the predicted load time provided by MySQL Autopilot.
Once the desired tables are loaded into the HeatWave cluster memory, customers can monitor the actual breakdown of the memory consumption with the detailed breakdown of each table’s in-memory footprint.

**Interactive Query Interface & Workload Monitoring**

The MySQL HeatWave console provides a query editor to ease the customer’s interaction with their database system by eliminating the need to go back and forth between the console and an external MySQL client for resource and data management, respectively. Customers can write and execute queries on MySQL HeatWave and view the query results through the query editor, while monitoring the state of the database, all in the same console.

In addition, MySQL HeatWave workload monitoring summarizes the results of the most recent queries executed along with their execution time and the
number of rows in the result set, allowing the customers to have access to the recent history of activities in the MySQL HeatWave database from the console.

**MySQL Autopilot Auto Shape Prediction Advisor**

MySQL Autopilot - Auto shape prediction continuously monitors OLTP workloads to provide suggestions that adapt to evolving workload patterns, allowing customers to maintain the best OLTP price-performance over time. It also shows trends on the workload with metrics such as buffer pool hit ratio and buffer pool utilization factor, providing insights for customers to better understand their workloads. The visual representation within the console makes it easy for database users to upsize or downsize their database shape to optimize price-performance.
Autopilot Indexing

Autopilot indexing recommends the right set of indexes for columns to improve OLTP query performance. It balances cost, storage space, and performance by adding or removing indexes.

Example of Autopilot Indexing showing (i) CREATE and DROP index suggestions, (ii) explanations for the recommendations.
Example of Autopilot Indexing showing (iii) estimated performance impact, and (iv) estimated storage impact from the recommendations

**Performance Monitoring**

The MySQL HeatWave console allows users to monitor the overall and “per-node” utilization of MySQL HeatWave hardware resources such as CPU, memory, and storage. It also provides a detailed breakdown of your resource consumption, such as data dictionary size, buffer pool size, and database connections.
HeatWave AutoML Model Creation, Prediction, and Explanation

The MySQL HeatWave console on AWS allows customers to train machine learning models, to use the trained models for predictions, and to inspect the explanations for the models and predictions. Users can tune the model by adjusting the training parameters (such as algorithms and features to include), specify different scoring metrics for model or prediction evaluation, and evaluate what-if scenarios by manipulating different feature values to understand how they affect the model's predictions. The console enables business analysts build, train, and run ML models without using SQL commands or any coding, simply using the visual interface.

Conclusion

Oracle believes in giving customers a choice. By making MySQL HeatWave natively available on AWS, customers can very easily benefit from the only cloud database service that combines transactions, analytics, and machine learning services into one MySQL Database, delivering real-time, secure analytics without the complexity, latency, and cost of ETL duplication—on AWS. Customers don’t have to face exorbitant data egress fees charged by AWS and higher latency when accessing a database service running in Oracle’s cloud. MySQL HeatWave on AWS is optimized for AWS with a superior architecture that delivers higher
performance and lower cost, as demonstrated by industry-standard benchmarks.

Try MySQL on HeatWave on AWS for free now!