ORACLE DevLive | Level Up | MySQL Summit

# Deep Dive for Machine Learning Development with MySQL HeatWave

**Sandeep Agrawal**
Consulting Member of Technical Staff
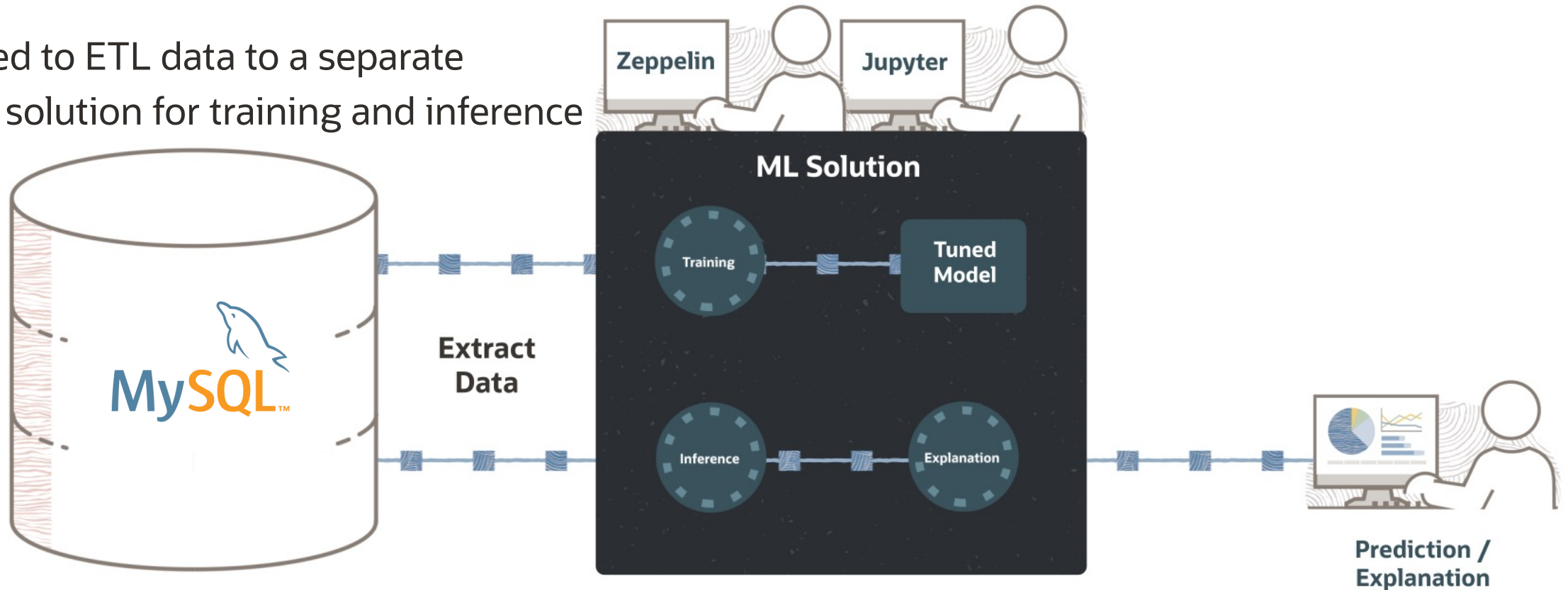Oracle

# In-Database ML

# Complexity of using external ML

Need to ETL data to a separate
ML solution for training and inference



Zeppelin

Jupyter

ML Solution

Training → Tuned Model

Extract Data

Inference → Explanation

Prediction / Explanation

- Complex, time-consuming
- Increases costs and risks
- Need to learn new tools/languages

**And it gets worse when using other databases...**

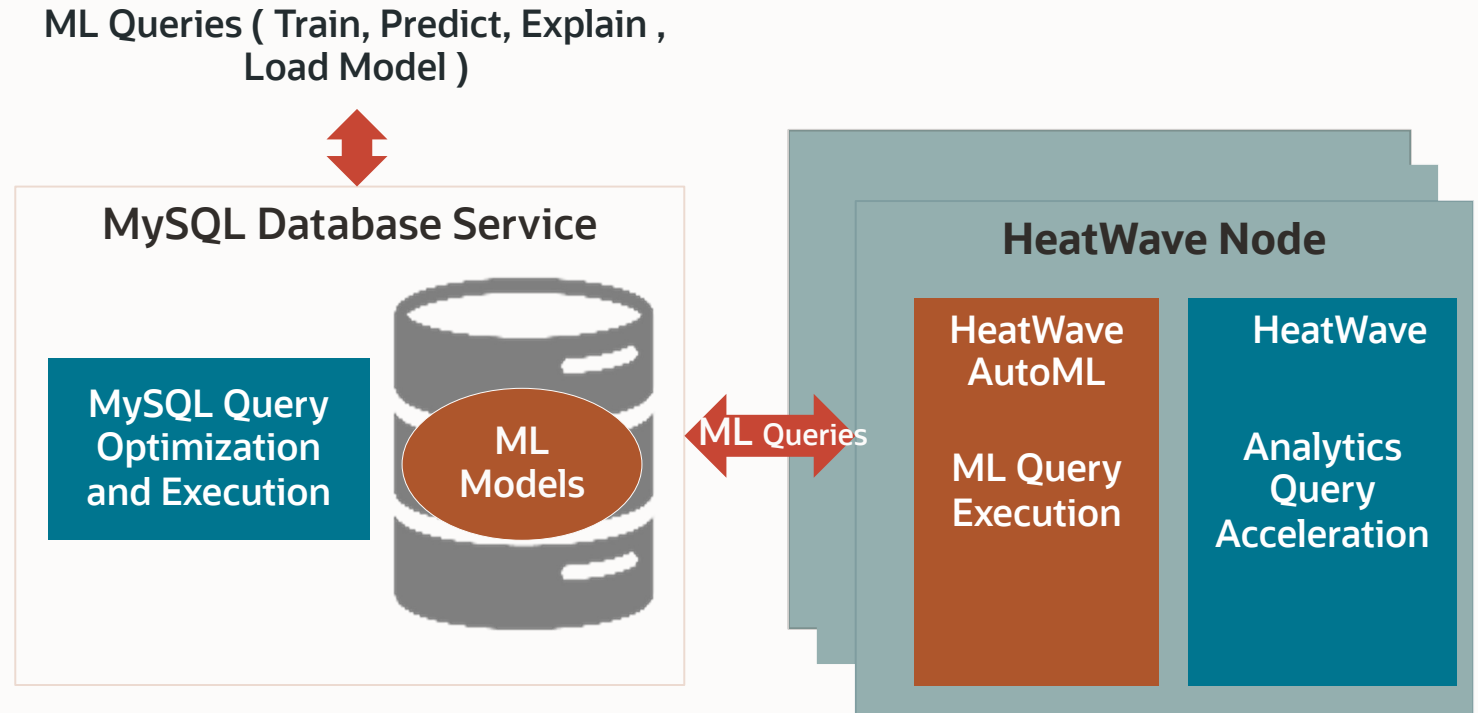# In-database Machine Learning with MySQL HeatWave
Fully automated, model and data don't leave the database, no additional cost
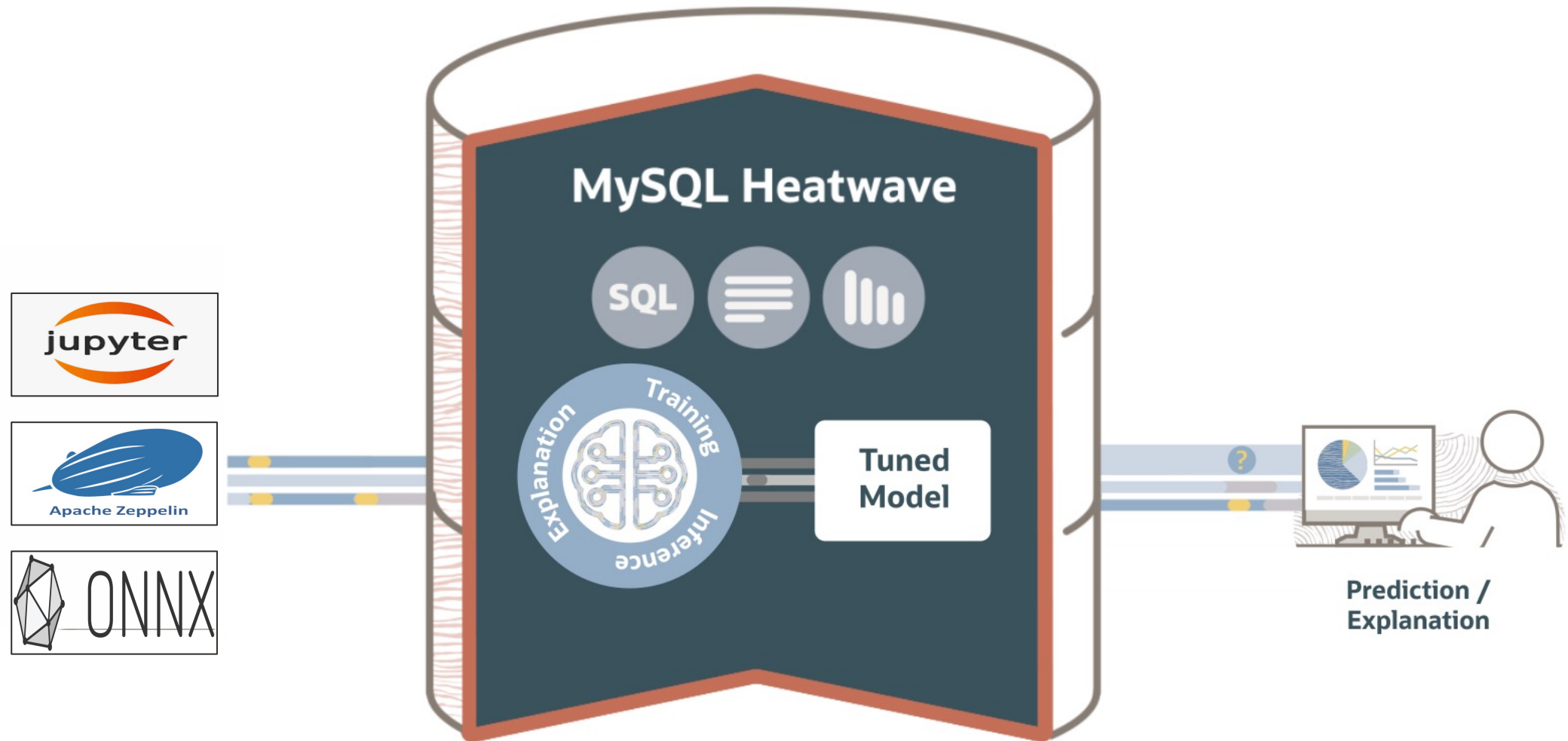
# Automated ML with MySQL HeatWave

# HeatWave AutoML

- Fully Managed, highly scalable, cost-efficient machine learning solution

- No need to move data or ML models outside of the Database Service

- Database-user friendly interfaces

- Leverages HeatWave cluster for distributed ML

- Model agnostic explanations for understanding both model and prediction behavior

**ML Queries ( Train, Predict, Explain , Load Model )**

**MySQL Database Service**

**MySQL Query Optimization and Execution**

**ML Models**

**ML Queries**

**HeatWave Node**

**HeatWave AutoML**

**ML Query Execution**

**HeatWave Analytics Query Acceleration**

# Target Use Cases and Personas

- Use Cases:
    - **Classification**: A discrete category is predicted for given input data
    - **Regression**: A continuous value is predicted for given input data
    - **Time Series Forecasting**: A trend is predicted into the future
    - **Anomaly Detection**: An anomalous value is predicted for a given input data
    - **Recommender System**: A set of recommended values is predicted for a given input data

- Personas:
    - **Database users**: Familiar and easy to use MySQL interface
    - **Data scientists**:  Bootstrap workflow with a good model, enable continuous experimentation

 3/30/23

# MySQL HeatWave AutoML Explanability
## Model and Prediction Explanations

- Model Explanations

  - Create during model training

  - Identifies most important features to the model

  - Shap, Fast Shap, Permutation Importance, Partial dependence plots

- Prediction Explanations

  - Identifies significant features for a few rows or an entire table
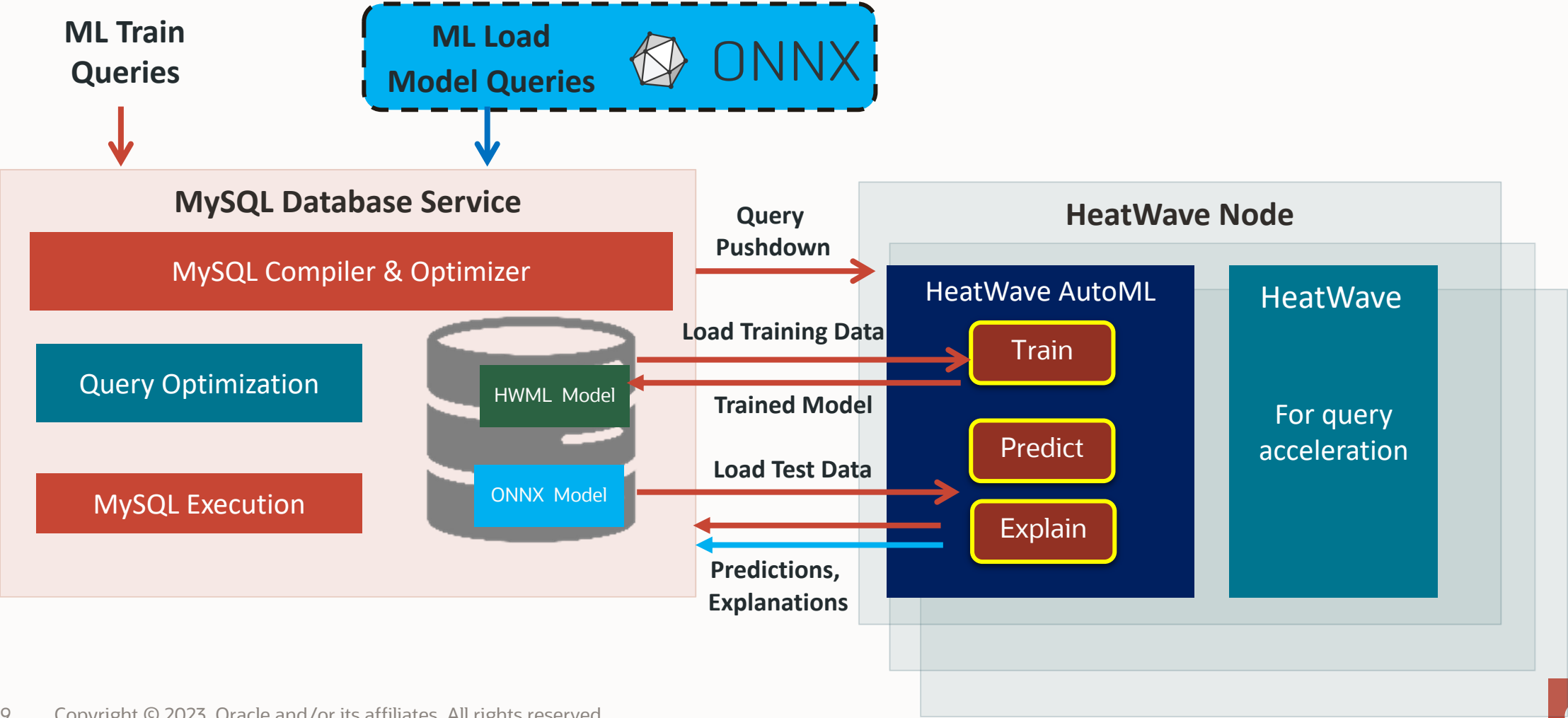
  - Shap, Permutation Importance
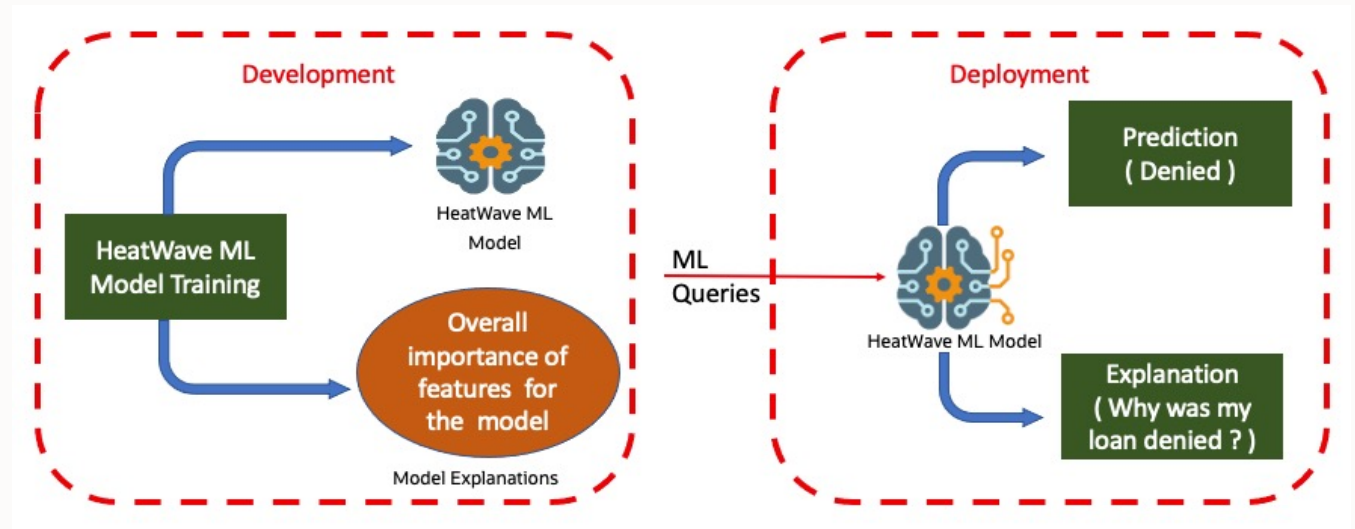


Evaluate Model: model20230126154033713
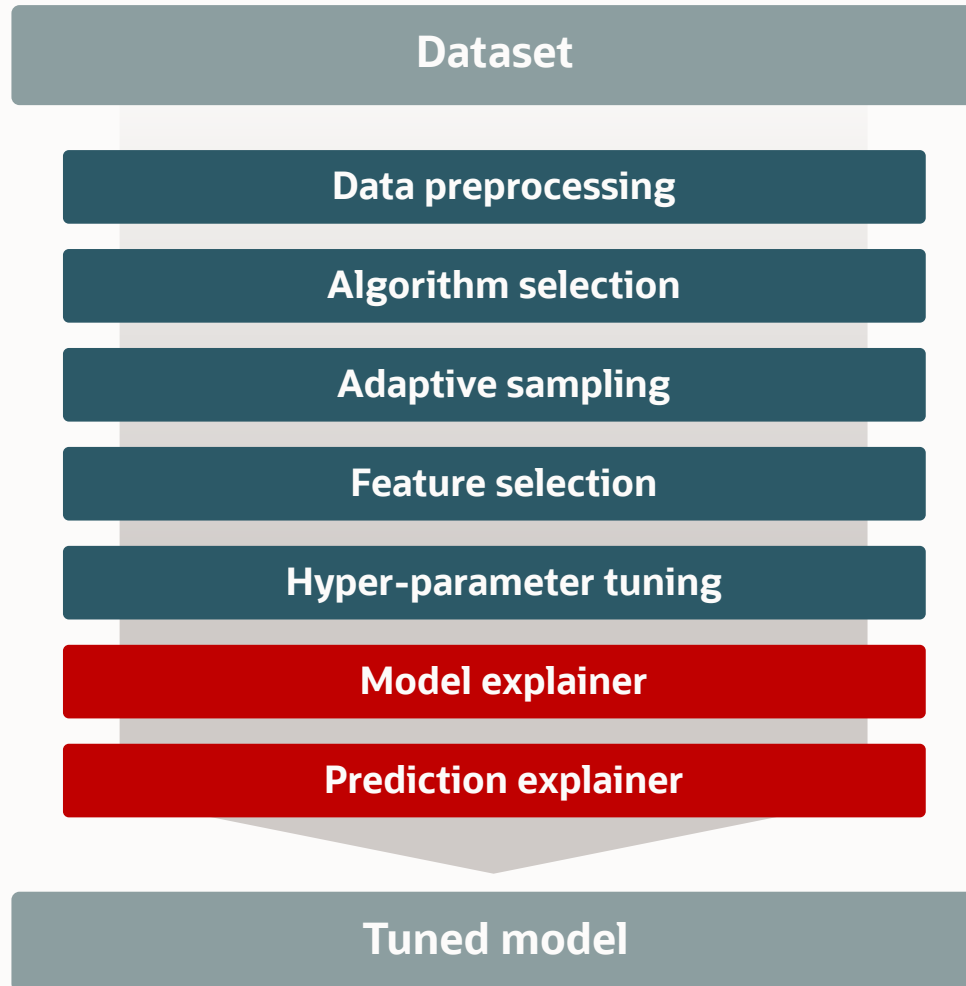
Model score   **Explain model**   Predictions

Feature Importance

| Name ⌄ | model20230126154033713 ⌄ |
|---|---|
| capital-gain | 0.0539 |
| education-num | 0.0366 |
| age | 0.0305 |
| relationship | 0.0254 |
| occupation | 0.0208 |

# Bring your own model (ONNX)



**ML Train Queries**

**ML Load Model Queries** ONNX

**MySQL Database Service**

MySQL Compiler & Optimizer

Query Optimization

MySQL Execution

HWML Model

ONNX Model

**Query Pushdown**

**Load Training Data**

**Trained Model**

**Load Test Data**

**Predictions, Explanations**

**HeatWave Node**

**HeatWave AutoML**

Train

Predict

Explain

**HeatWave**

For query acceleration

# MySQL HeatWave AutoML automates the ML lifecycle

Dataset

Data preprocessing

Algorithm selection

Adaptive sampling

Feature selection

Hyper-parameter tuning

Model explainer

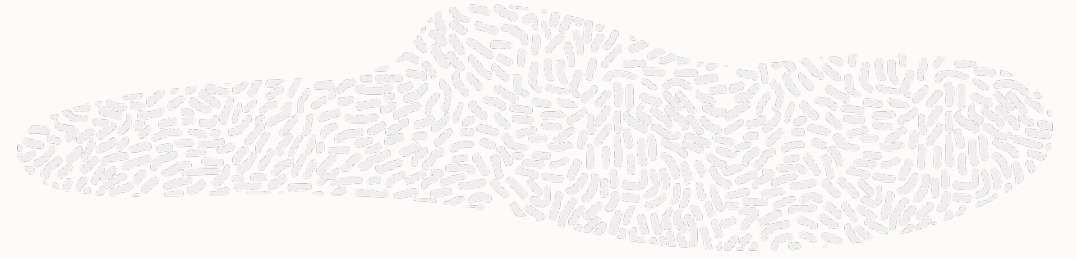Prediction explainer

Tuned model

Regulatory compliance

Fairness

Repeatability

Causality

Trust

# HeatWave AutoML vs Redshift ML

✓ Trains models **25X faster** on average

✓ **1% of the cost**

✓ **Scales** as more nodes are added

See Benchmark details: https://www.oracle.com/mysql/heatwave/performance/

# MySQL HeatWave AutoML provides SQL interfaces
## Database user friendly

CALL sys.ML_TRAIN('Census.census_train', 'revenue', JSON_OBJECT('task', 'classification'), @census_model);

CALL sys.ML_MODEL_LOAD(@census_model, NULL);

CALL sys.ML_SCORE('Census.census_test', 'revenue', @census_model, 'balanced_accuracy', @score);

SELECT sys.ML_PREDICT_ROW(@row_input, @census_model);

SELECT sys.ML_EXPLAIN_ROW(@row_input, @census_model, NULL);

# Product Demo