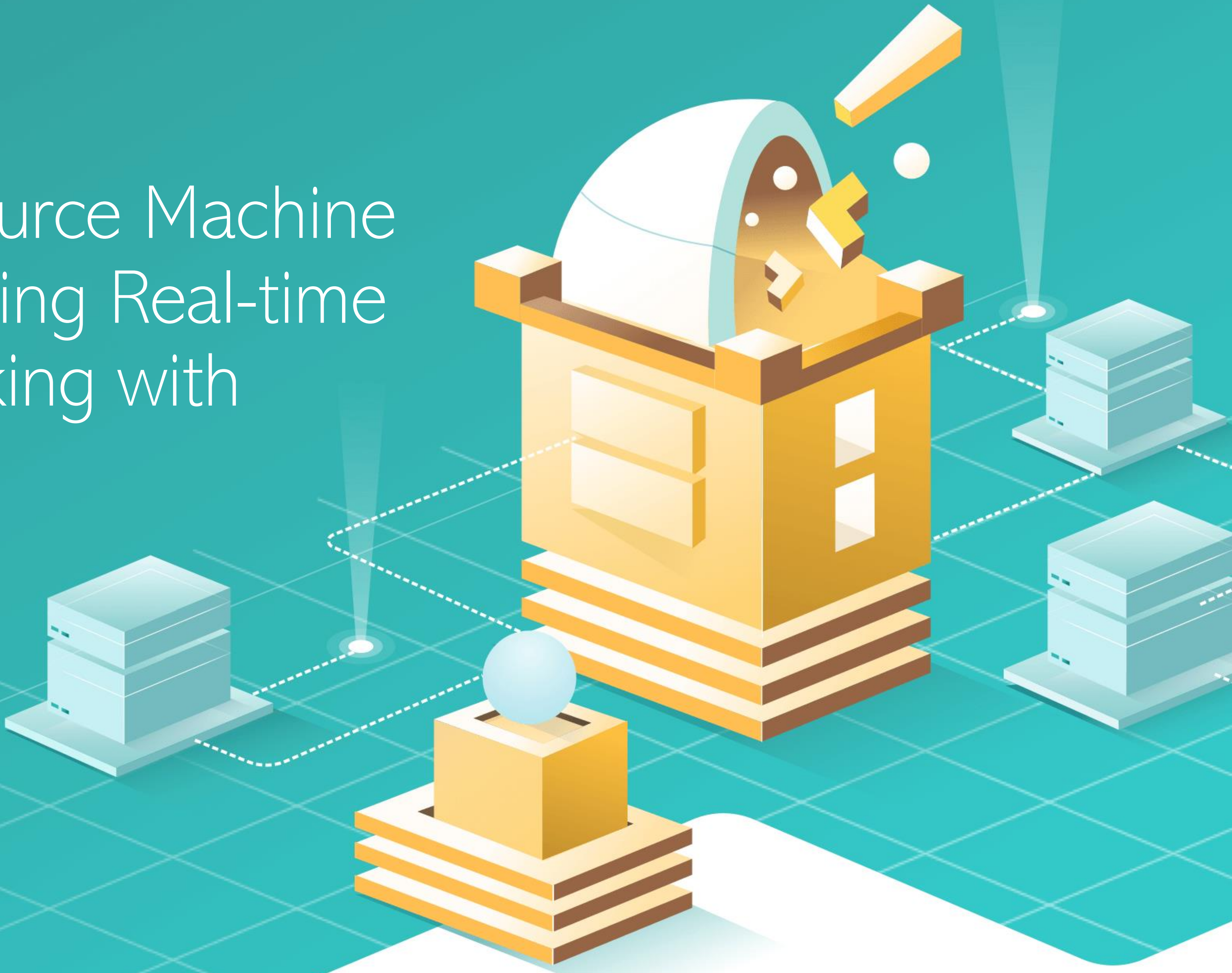
OpenMLDB

OpenMLDB : Open-source Machine Learning Database Driving Real-time Intelligent Decision-making with Real-time Features



Lu Mian

OpenMLDB PMC

Jan 2024

About Me

Lu Mian

lumian@4paradigm.com

- OpenMLDB Head of R&D
- System Architect at 4Paradigm
- Ph.D from HKUST

Contents

OpenMLDB

1. Engineering Challenges in Real-Time Intelligent Decision Making

Real-time Intelligent Decision-making based on Machine Learning Requires Millisecond Level Real-time Computing Capability

Two major AI applications: Perception-based AI and **Decision-based AI**

Only hard real-time computing can truly meet real-time decision-making requirements - **Real-time Data, Real-time Computing**

Streaming computing are mostly designed for Big Data and BI

Minutes/ Hours	**Seconds**	**Milliseconds**
Batch	Streaming	Hard Real Time

python Spark Streaming Flink

R sas APACHE STORM kafka

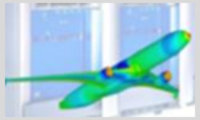
TensorFlow

The AI real-time computing in the market nowadays is mostly streaming computing

AI Unmanned Vehicle

Anti-fraud

Quantitative Trading



Aerospace

Bank requires millisecond-level responses

Taking a certain bank's anti-fraud scenario as an example

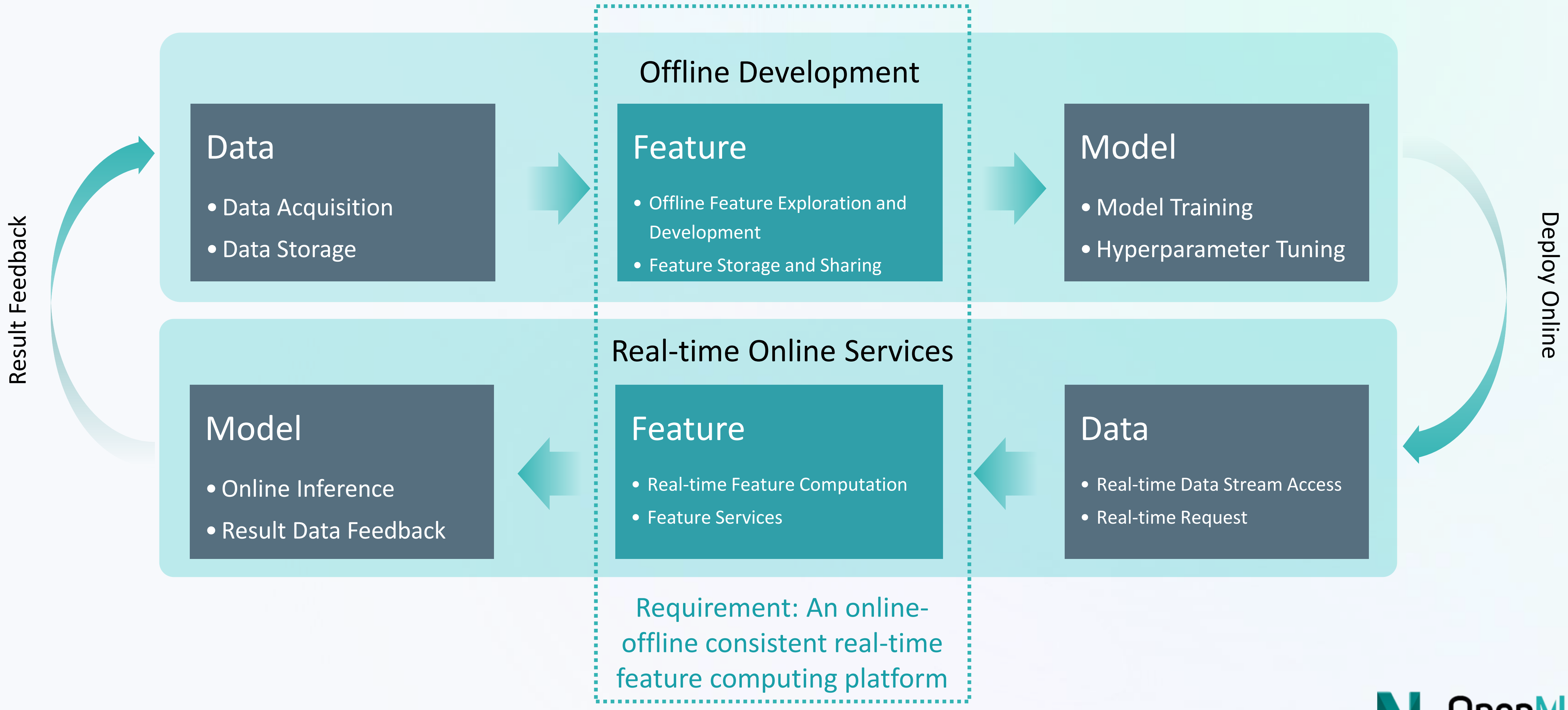
Customer Requirement: A highly-accurate anti-fraud system with a response time of 20ms for feature computation

Solution	Respone Time	Precision and Recall
Traditional Rule System	~200ms	Relatively Poor
Customer Self-Developed System	~50ms	Moderate
The Prophet of the Fourth Paradigm	<20ms	Excellent

Hard real-time scenarios contain enormous commercial value, and there are few universal commercial products available

OpenMLDB

Intelligent Decision-making based on Machine Learning from Offline Development to Online Deployment



Real-time Feature Computation for Anti-fraud Transactions

Swiping Records

Card Number	Swiping Amount	Swiping Time
012159	1000	2022/01/12 08:00:00

Virtual Insertion

Transaction History Table

Card Number	Swiping Amount	Swiping Time (Sorted)
012112	223	2022/01/12 02:00:00
012159	15	2022/01/12 06:00:00
012159	1000	2022/01/12 07:59:55
012159	2000	2022/01/12 07:59:57
012159	1000	2022/01/12 08:00:00

Window Aggregation

3h

10s

Feature Computation

Generated Features

Card Number	Swiping Amount	In the past 10 seconds: Number of card swipes | Maximum Amount Swiped | Minimum Amount Swiped | Average Amount	In the past 3 hours: Number of card swipes | Maximum Amount Swiped | Minimum Amount Swiped | Average Amount
012159	1000	3 | 2000 | 1000 | 1333	4 | 2000 | 14 | 1003

Model Inference

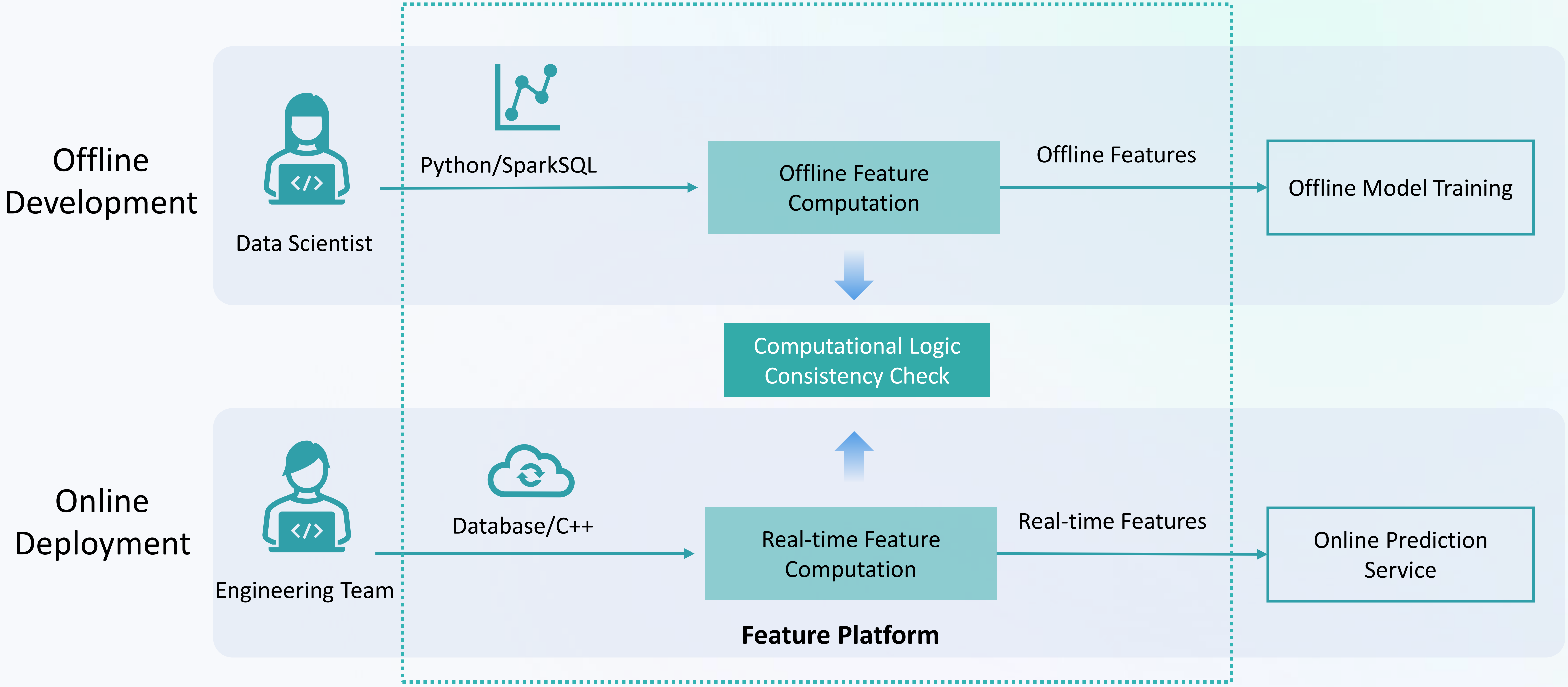
Fraudulent Transactions?

Engineering Requirements

1. Online and offline consistency
2. Low latency, high concurrency, and high availability

OpenMLDB

Traditional Feature Development: Separation of Offline Development and Online Deployment Incurring High Investment Cost



OpenMLDB

Possible Reasons for Online and Offline Inconsistency

Inconsistency in tool capabilities

Offline Development

Python

```python
import pandas as pd
t1 = pd.read_csv("data.csv")
account_feat = t1['account'].std()
```

$$\sqrt{\sum_{i=0}^{N} \frac{(x_i - \mu)^2}{N-1}}$$

standard deviation
(Bessel's Correction)

Online Deployment

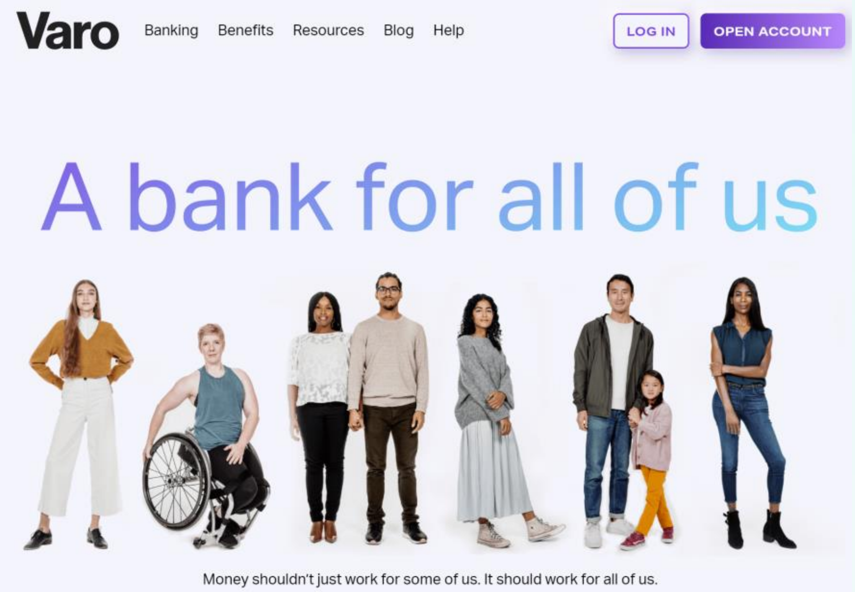
MySQL

```sql
select std(account) as account_feat from t1;
```

$$\sqrt{\sum_{i=0}^{N} \frac{(x_i - \mu)^2}{N}}$$

standard deviation

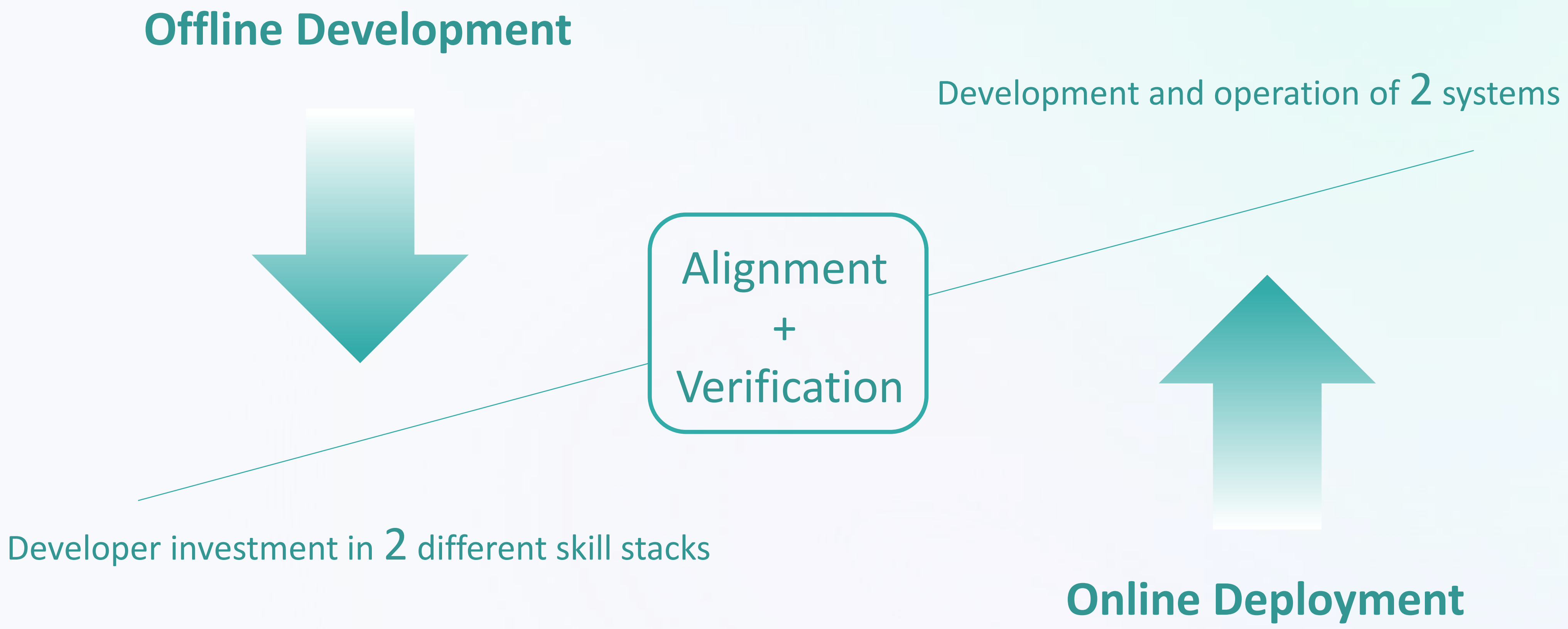
Different understanding of demands



	Account Balance
Online Deployment	current "account balance"
Offline Development	"account balance" as of yesterday

OpenMLDB

The High Cost of Engineering Implementation Brought About by Online and Offline Consistency Verification



OpenMLDB

2. OpenMLDB Provides Online-Offline Consistent Real-Time Feature Computation

OpenMLDB

OpenMLDB Development History: From Close-source to Open-source

Past 5 Years	RTIDB/FEDB (4Paradigm Close-source)	2021.6 →	OpenMLDB (Open-source/commercialization)

Before Open-source, brought out within 4paradigm SageOne and implemented in over 100 scenarios, covering over 300 nodes.

After Open-source, actively embraces community developers with an open attitude, integrates the open-source ecosystem, and provides commercial customization and support.

akulaku 37GAMES JDT京东科技 HUAWEI VIPSHOP UBiX

Main Usage Scenarios

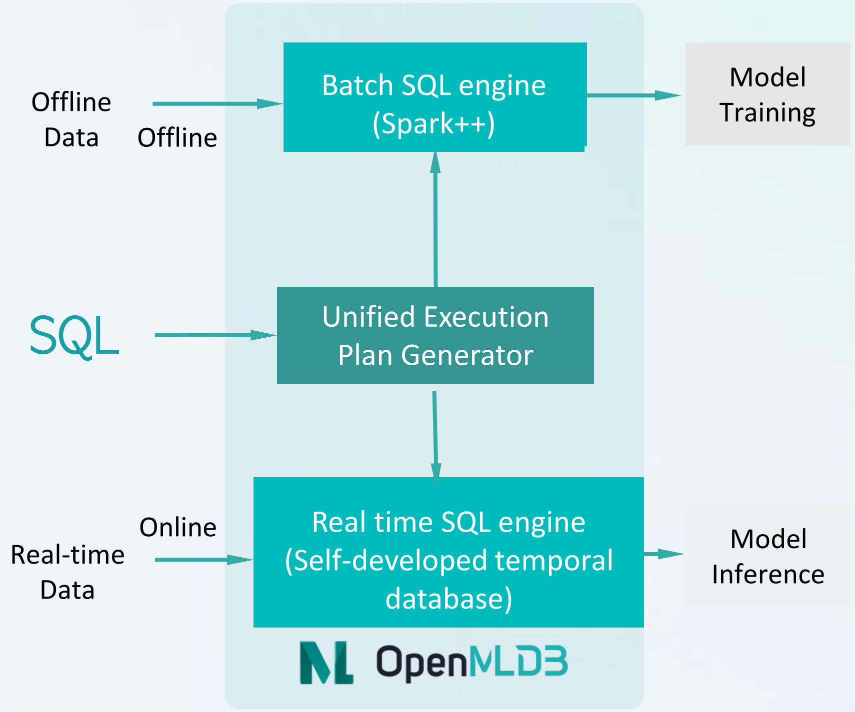
Credit Card Cash Installment Precise Marketing	Pre-loan Risk Scoring	Marketing Customer Acquisition	Personalized Recommendation	Intelligent Identification of Suspicious Transactions in Anti-money Laundering
Credit Card Account Risk Warning	Compliance Quota Decision-making	Risk Management	Exploration of Investment Advisory Clients	Credit Card Application Anti-fraud
Fraud and Card Retention Prevention and Control	Personalized Financial Recommendations	Anti-fraud Measures for Retail Loans	Historical Customer Activation	Customer Loss Warning
Network Traffic Prediction	Transaction Fraud Scoring	Cash Installment Personalized Recommendation	Credit Card Transaction Anti-fraud	Financial Product Recommendations

OpenMLDB: Open-source Machine Learning Database, A Consistent Feature Platform both Online and Offline

Original Process

- Step 1: Feature script development — Scientist
- Step 2: Refactoring to meet online low latency, high throughput, and high availability — Engineer
- Step 3: Online and offline consistency verification — Scientist and Engineer
- Step 4: Production launch — Engineer

OpenMLDB Abstract Architecture



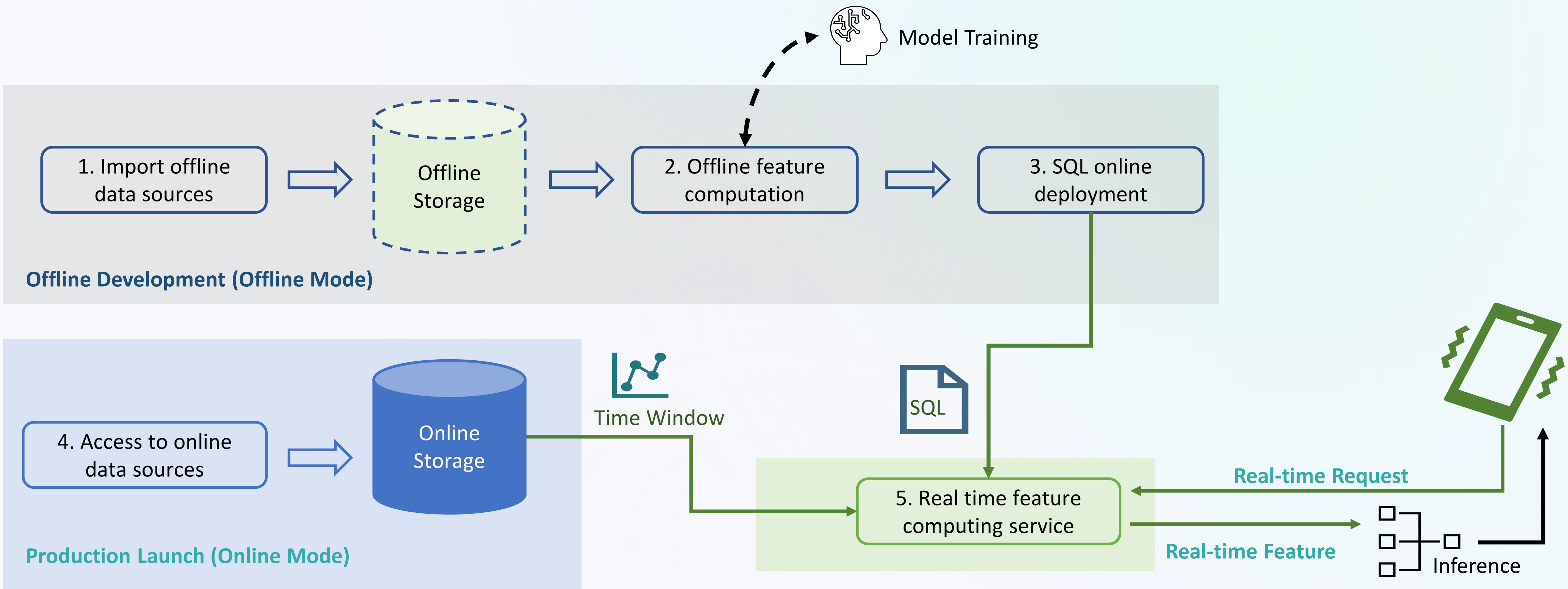
Process based on OpenMLDB

- Step 1: Feature script development — Scientist

Using OpenMLDB to achieve instant deployment and save months of manpower costs

- Step 2: One-click production launch — Scientist

Complete Process from Offline Development to Online Services

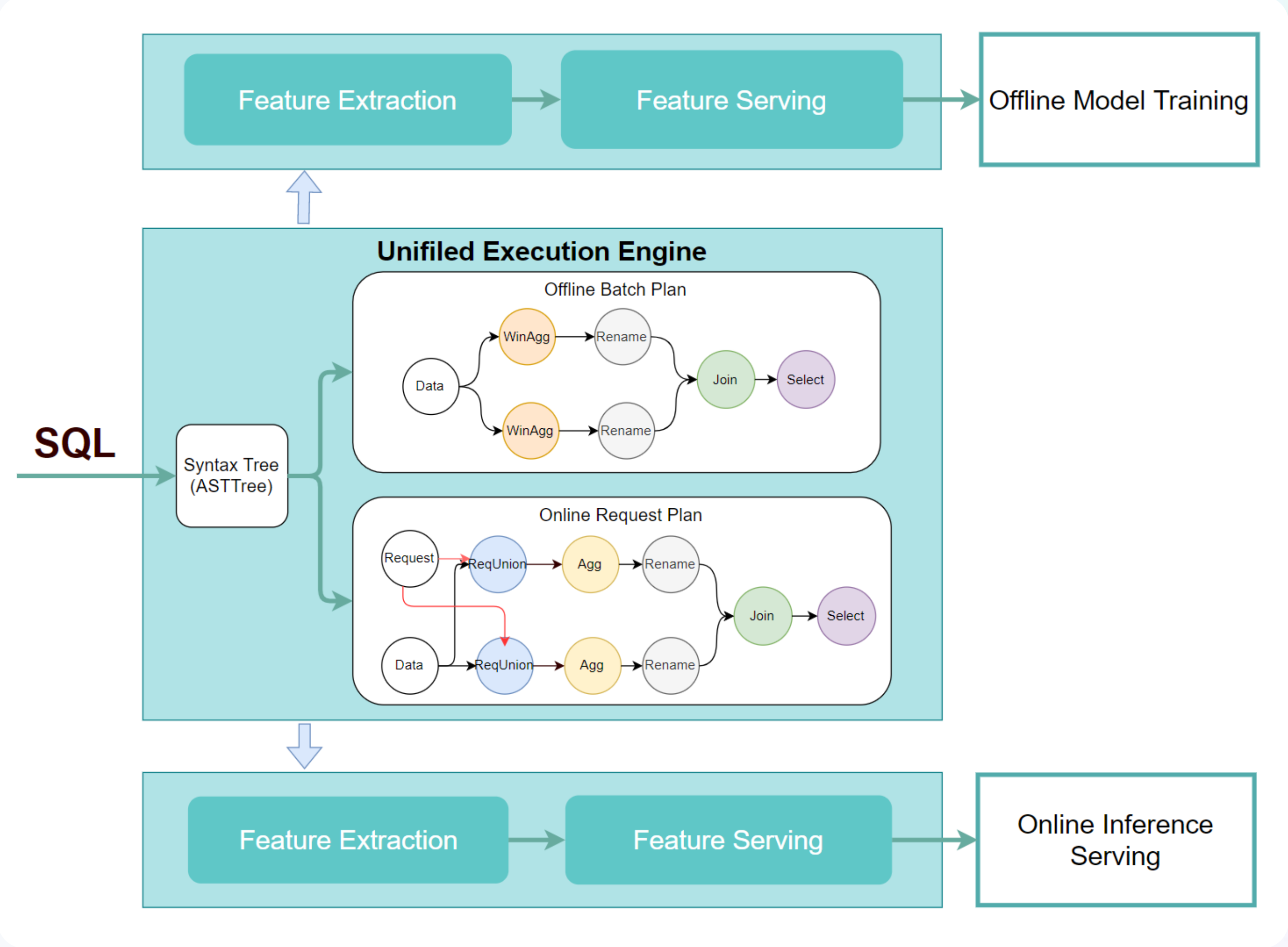


OpenMLDB

OpenMLDB provides a Consistent Millisecond Level real-time feature computing platform both Online and Offline

- On-demand computation based on real-time data
- Defining features based on SQL
- Production-level platform, distributed, scalable, and highly-available

Core Component 1: Online and Offline Consistent Unified Execution Engine



- Unified underlying computing functions
- Adaptive adjustment of online and offline execution modes from logical plan to physical plan

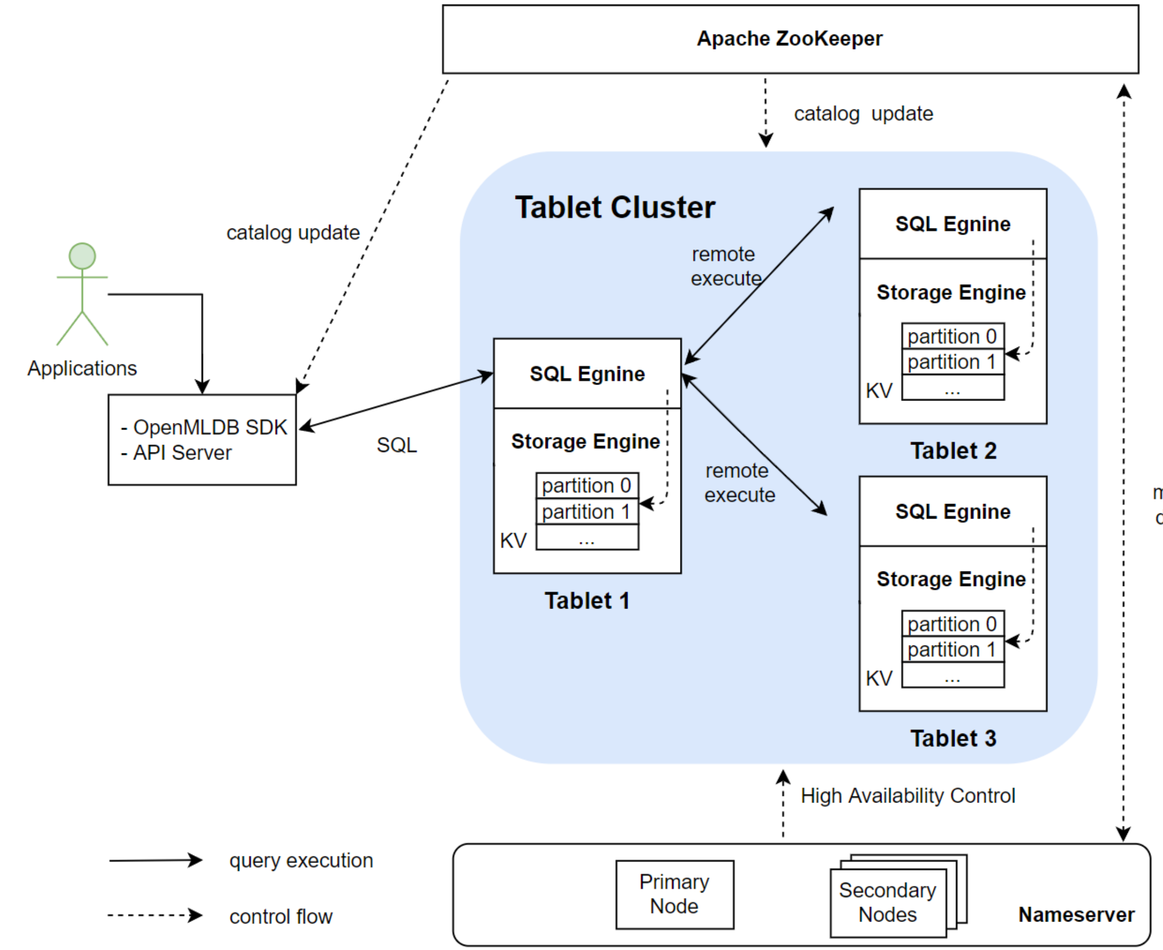
Consistency between online and offline is naturally guaranteed

Core Component 2: High Performance Real-time SQL Engine

Main modules of distributed real-time SQL engine

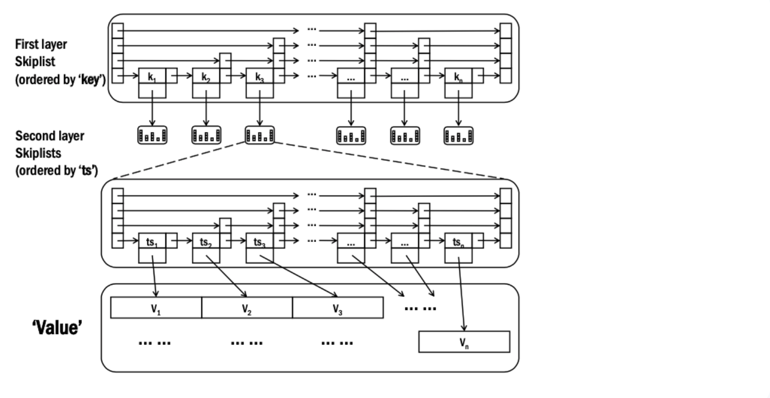
- **ZooKeeper** - Metadata storage and management
- **Nameserver** - Tablet Management and Failover
- **Tablets**
  - Distributed SQL Execution Engine
  - Distributed storage engine:
    dual storage engine for memory and disk
- High performance, scalability, and high availability

For a detailed description of the online engine architecture, please refer to:
https://openmldb.feishu.cn/wiki/wikcnavULzxKH5Aka3ox0871R2f

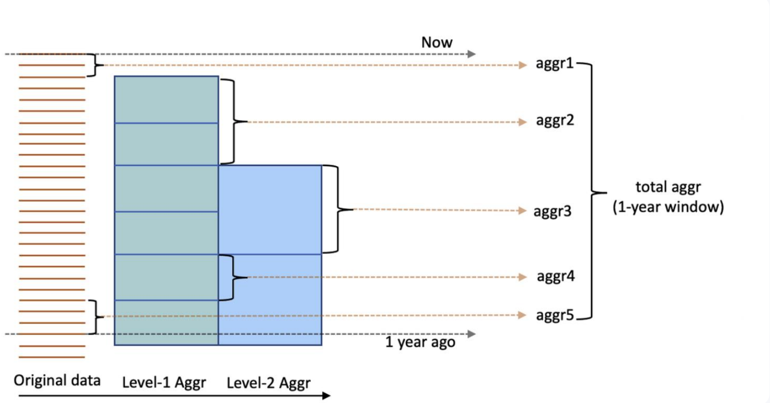


OpenMLDB

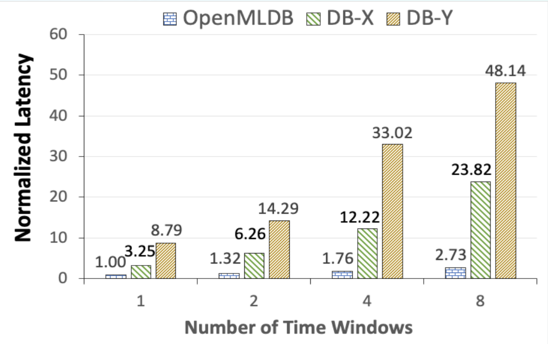
Core Component 2 (Continued): High Performance Real-time SQL Engine - Core Optimization Technology

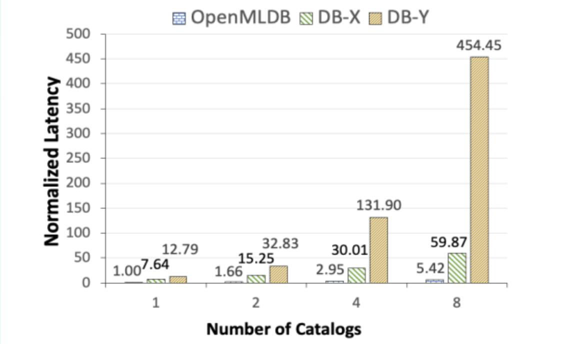


Core optimization: Double layer skip list (memory engine)

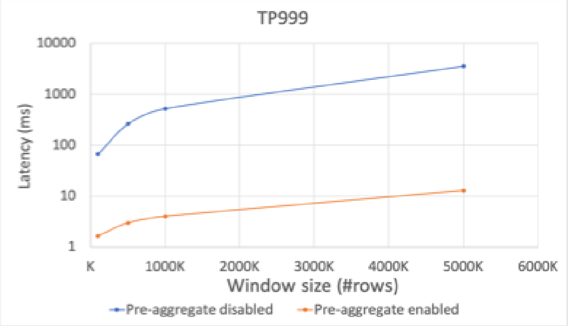


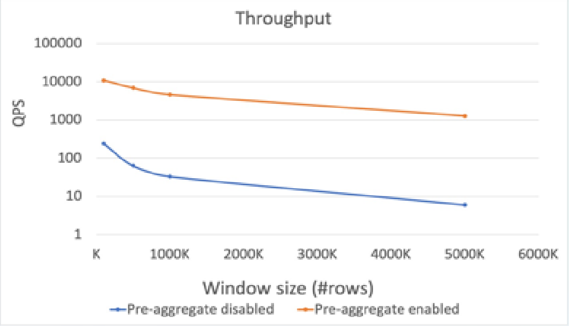
Core optimization technology: Pre-aggregation





Real-time computing performance: comparison between OpenMLDB and other commercial databases



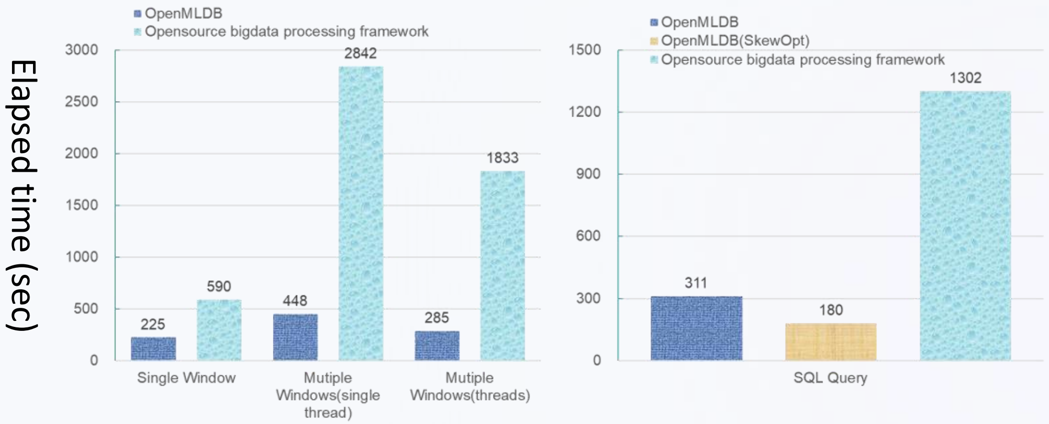


Performance improvement of OpenMLDB using pre-aggregation technology

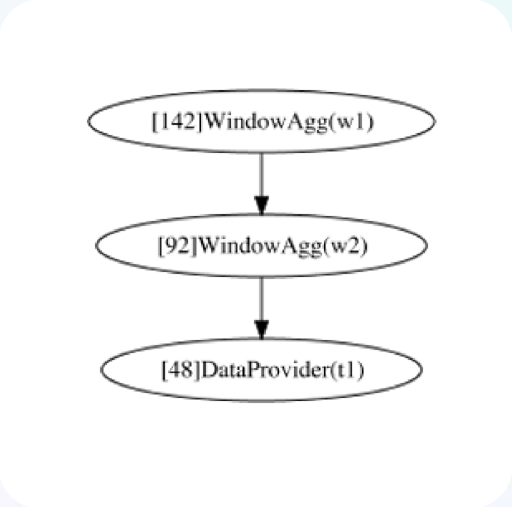
The complete test report: https://openmldb.feishu.cn/wiki/wikcnZRB9VRkqgD1vDFu1F9AaTh

Core Component 3: An Optimized Offline Computing Engine for Feature Computing

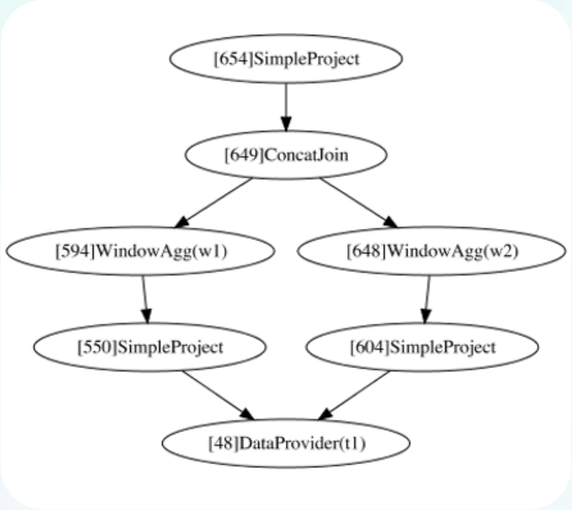
- Multi-window parallel computing optimization
- Optimization of data skew computation
- SQL syntax extension
- OpenMLDB Spark distribution optimized for feature computing




```sql
SELECT
  min(age) OVER w1 as w1_min_age,
  min(age) OVER w2 as w2_min_age
FROM t1
WINDOW
  w1 as (PARTITION BY name ORDER by age ROWS BETWEEN 10 PRECEDING AND CURRENT ROW),
  w2 as (PARTITION BY age ORDER by age ROWS BETWEEN 10 PRECEDING AND CURRENT ROW)"
```



Spark 3.0.0

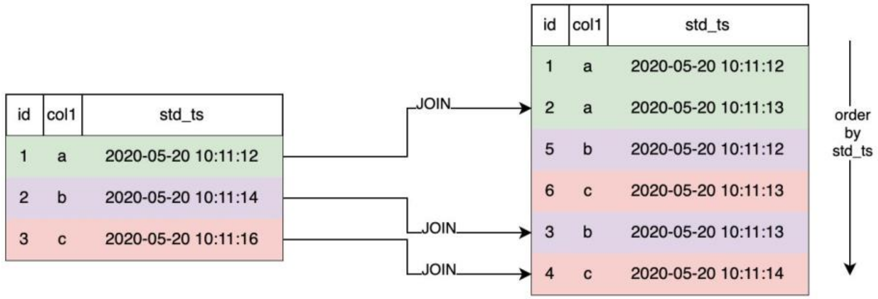


OpenMLDB

Core Component 4: SQL Extension for Feature Engineering

LAST JOIN

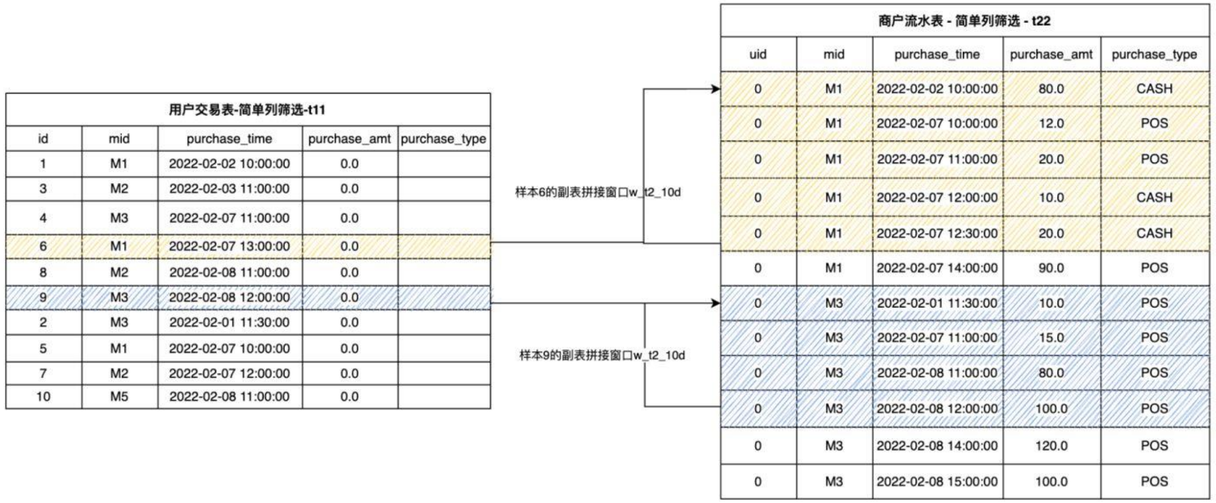
When matching multiple rows, only matches the latest record



id	col1	std_ts	id	col1	std_ts
1	a	2020-05-20 10:11:12	2	a	2020-05-20 10:11:13
2	b	2020-05-20 10:11:14	3	b	2020-05-20 10:11:13
3	c	2020-05-20 10:11:16	4	c	2020-05-20 10:11:14

WINDOW UNION

Cross table join and window aggregation operations (point-in-time) to avoid feature traversal

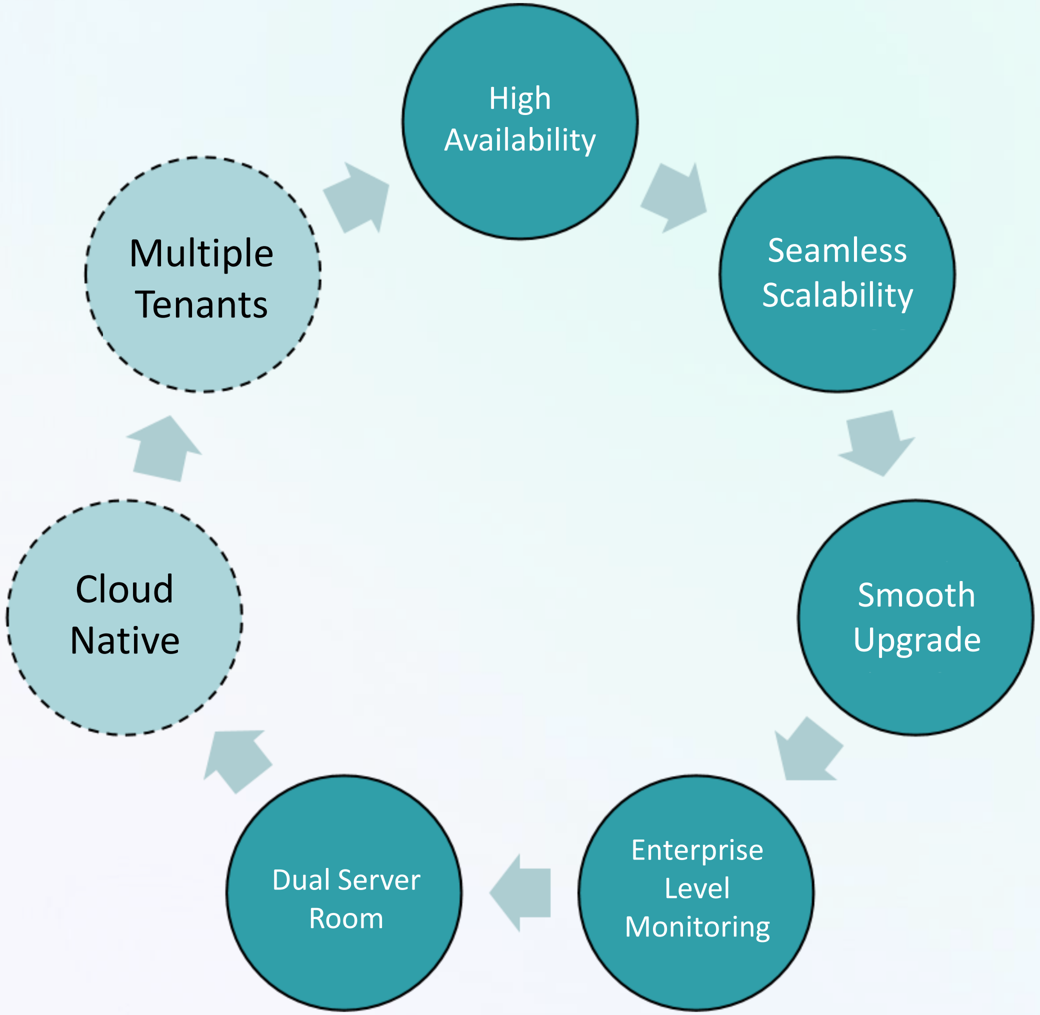


OpenMLDB

Enterprise Level Feature Support

Born for large-scale Enterprise Level applications

Implemented in Hundreds of Scenarios

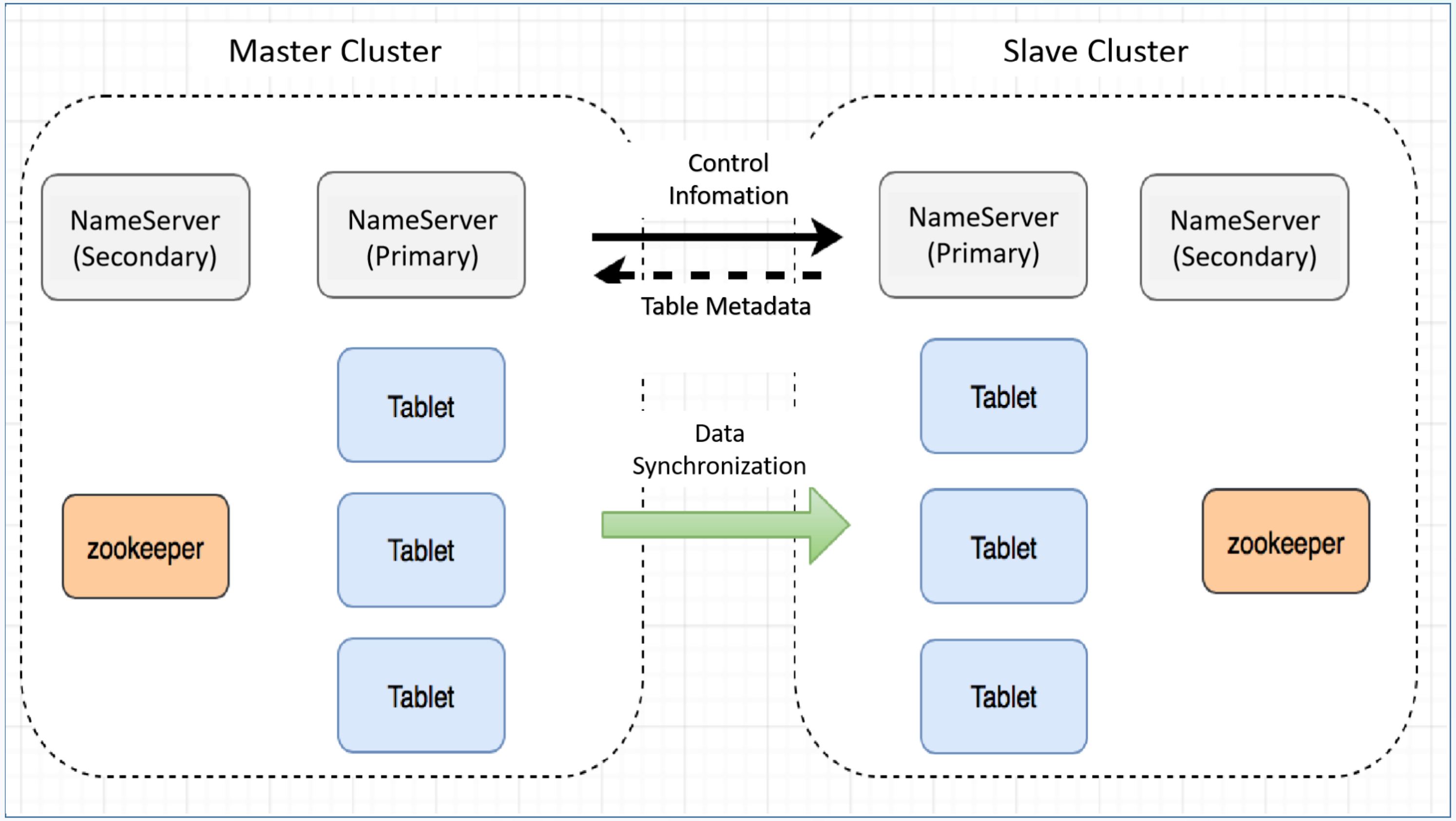


Advanced Feature: Cross Data Center Disaster Recovery

Deploy multiple OpenMLDB clusters in master-slave replication mode to achieve disaster recovery

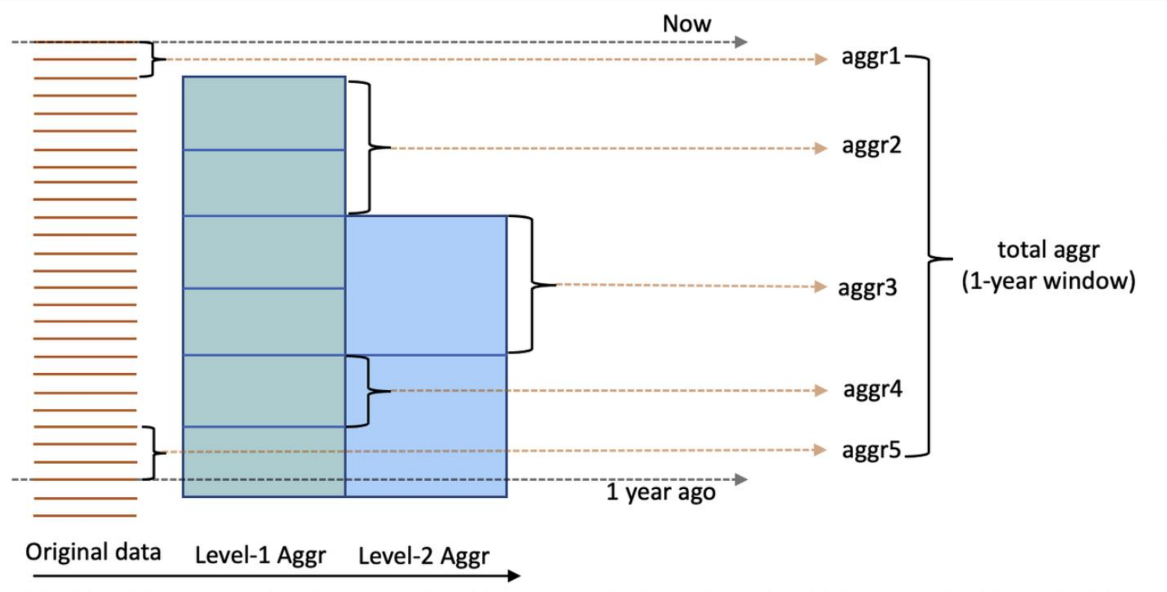
Master cluster: A cluster that can support read and write, and can synchronize data with the slave cluster

Slave cluster (one or more): A cluster that only serves read requests, with data consistent with the master cluster

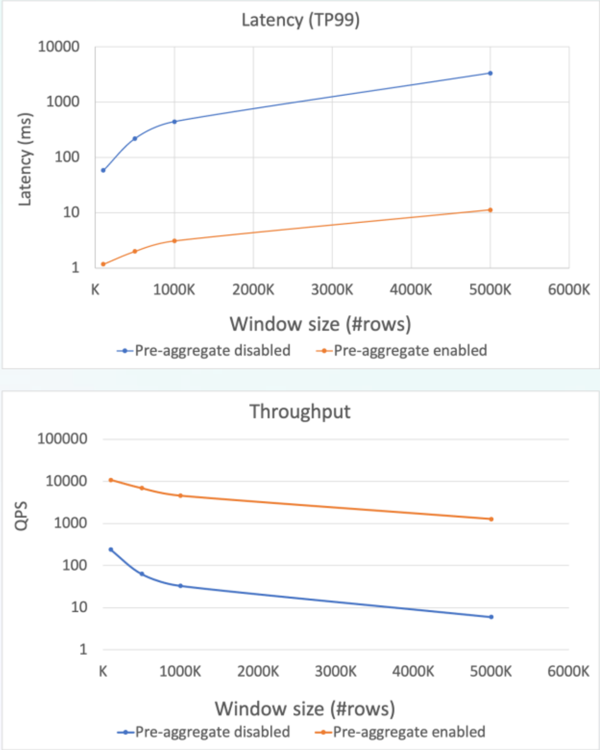


Advanced Features: Pre-aggregation

When the amount of data in the window is huge (millions), enabling pre-aggregation significantly optimizes the latency



Real-time window computation with a span of one year based on a two-layer pre-aggregated table

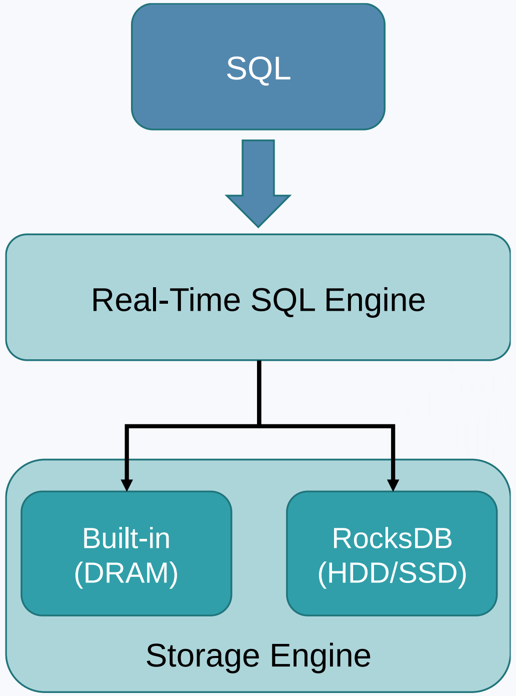


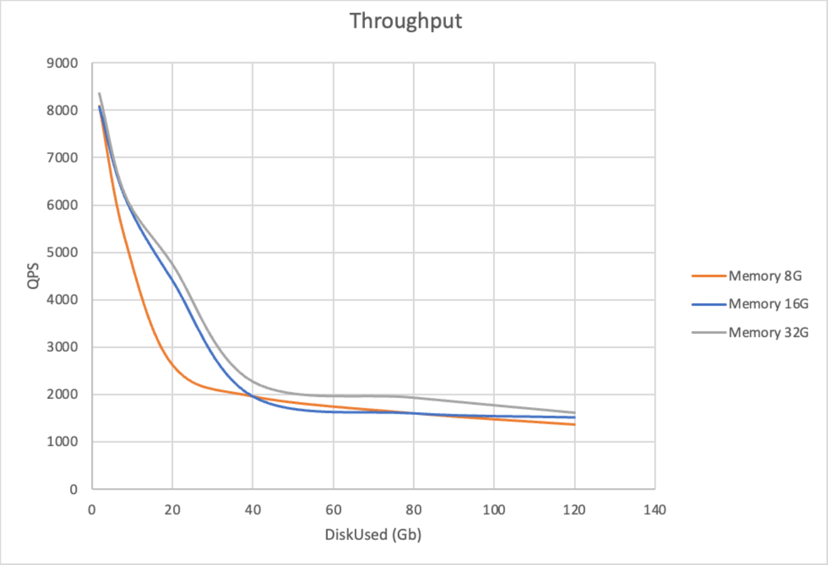
Pre-aggregation technology improves performance by 1-2 orders of magnitude

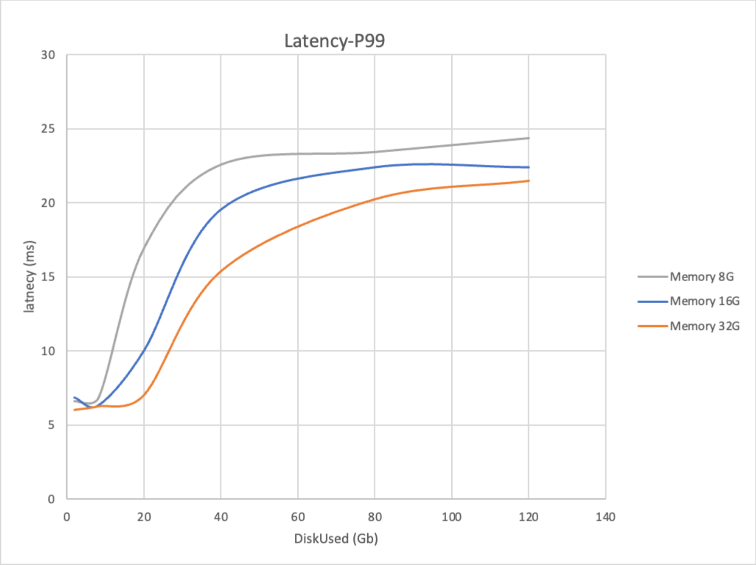
OpenMLDB

Advanced Features: Dual Storage Engine Architecture

The disk-based storage engine of the online engine provides a cost-saving solution that is more than 5 times that of the memory-based storage engine, but it will bring certain performance degradation. In practical scenarios, performance and cost requirements should be considered when selecting a solution





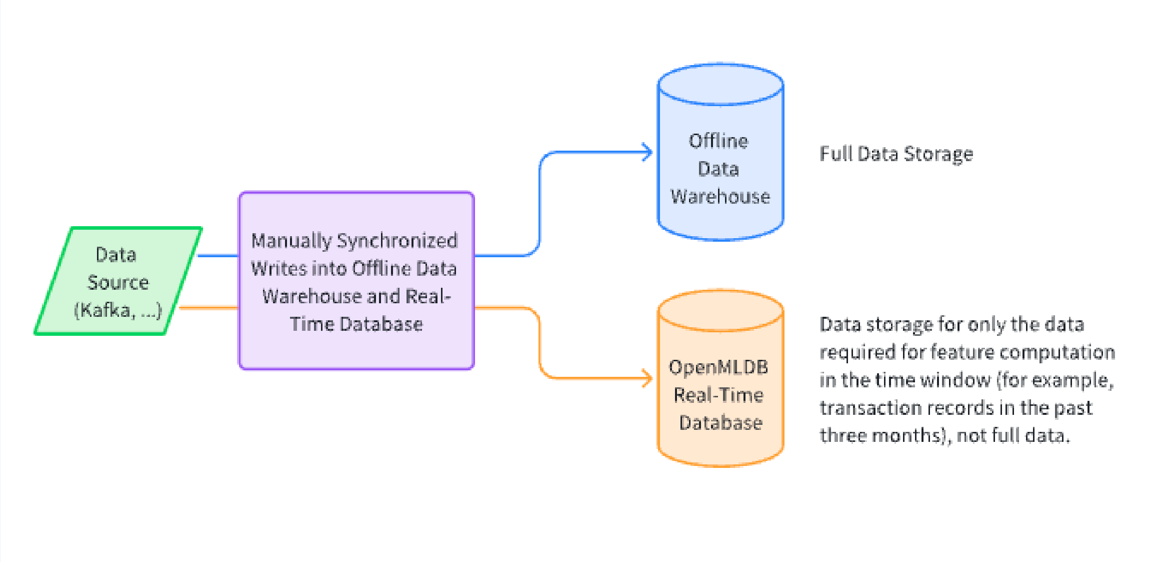


QPS/TP99 both have a performance degradation of 4-5x.
Choose the architecture depending on the application scenario.

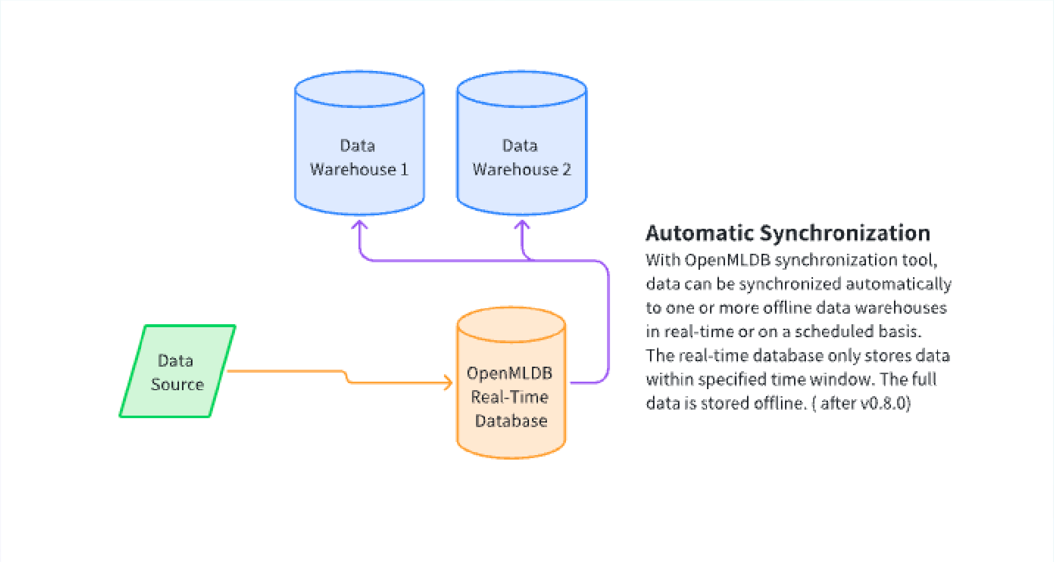
OpenMLDB

Advanced Features: Automated Online and Offline Data Synchronization

Online and offline data source synchronization, evolved from manual synchronization mechanism to automated synchronization mechanism



Manually establishing offline and online data synchronization



Automated online to offline data synchronization (Available after v0.8.0)

OpenMLDB

Advanced Features: Intelligent Operation and Maintenance Diagnostic Tool

Intelligent diagnostic tools: Automated service exception status check, intelligent log collection, …

Intelligent operation and maintenance tool: *openmldb_ops*, one-click data and table recovery, automatic scaling and sharding balance, …

Service Status Check

Automated abnormal state diagnosis
(Version verification, configuration file check,...)

One-click Data Recovery

Rapid data recovery under abnormal conditions
(Network exception, auto_failover=false,...)

Automatic Balance

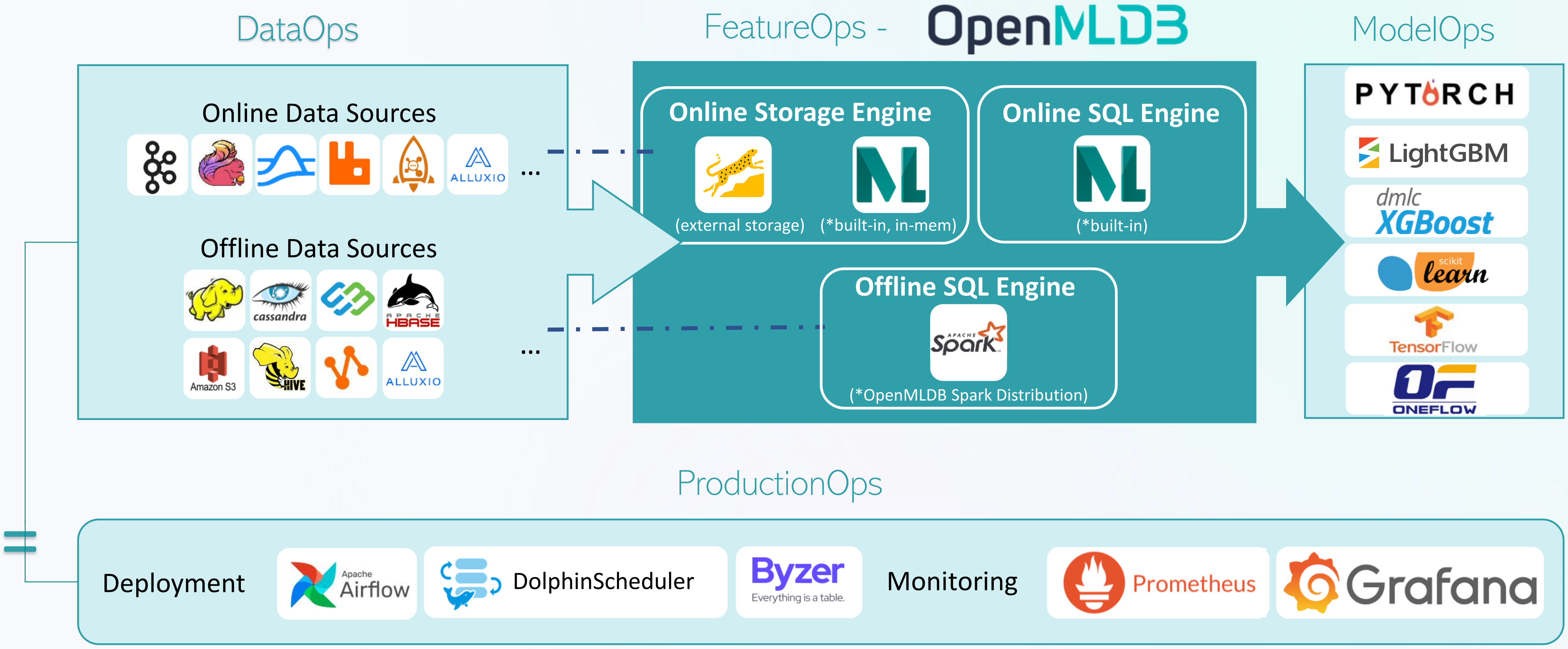
Automatic migration and rebalancing of shards after expansion and contraction
(Based on heuristic algorithms)

OpenMLDB

Advanced Features: Multi Form Deployment Support

3. Community Eco-System and Use-Cases

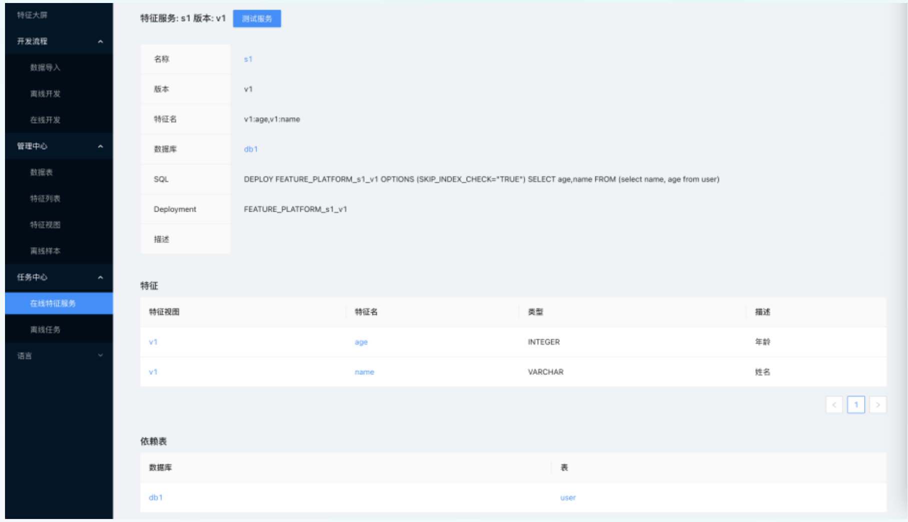
OpenMLDB Upstream and Downstream Open-source Ecosystem

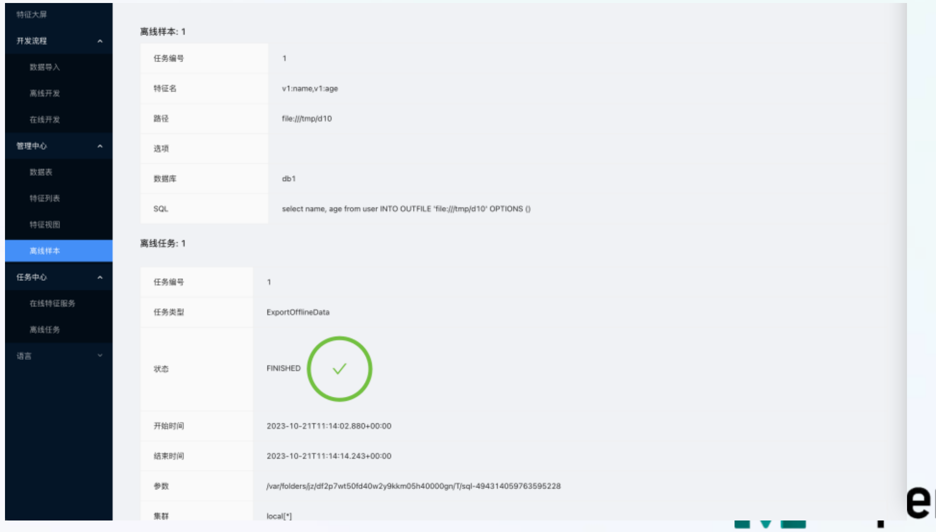


OpenMLDB Application Ecosystem - FeatInsight Feature Platform

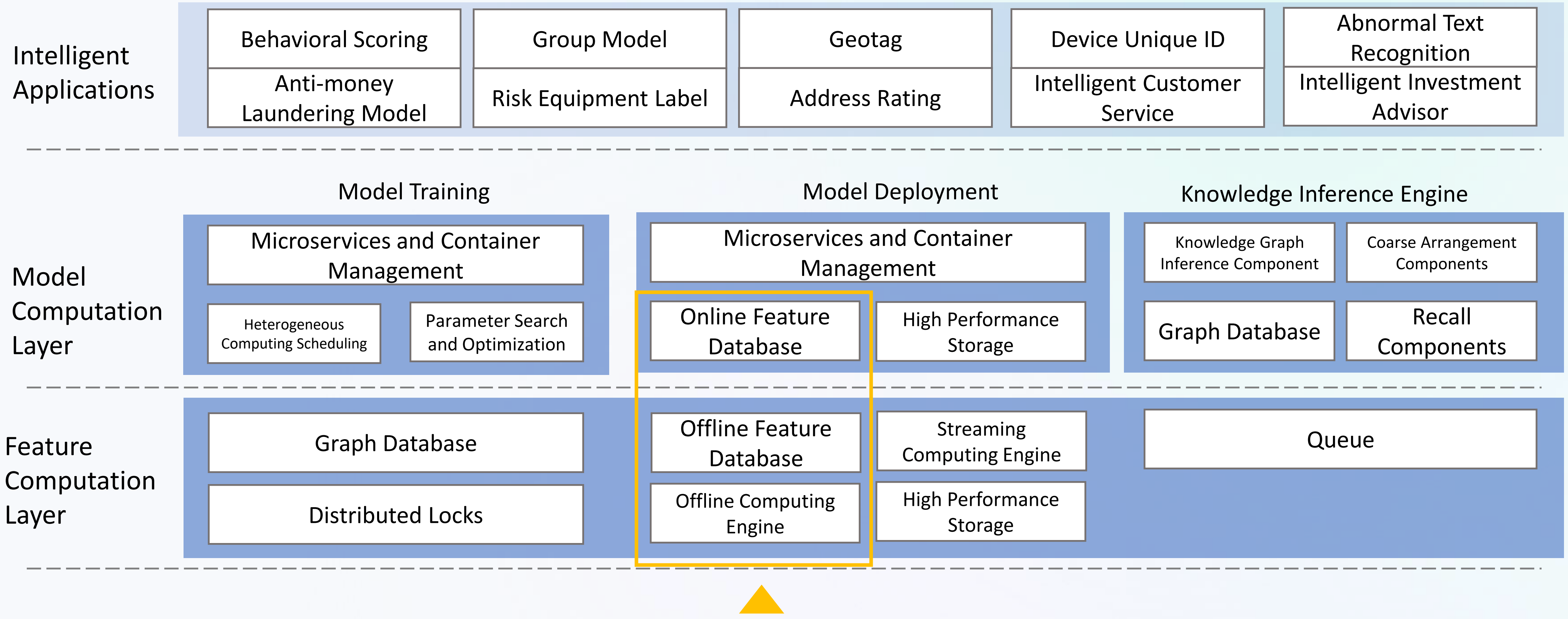
- Visual feature development and management interface
- Development Assistance for Large Complex Features Based on DAG
- Flexible feature reuse
- Feature lineage management and version management
- Millisecond-level real-time feature and pre-computed feature support

Beta version has been released:
https://github.com/4paradigm/FeatInsight





OpenMLDB Usecase - Feature Platform in Akulaku Intelligent Computing Architecture



Scenario-Driven: OpenMLDB

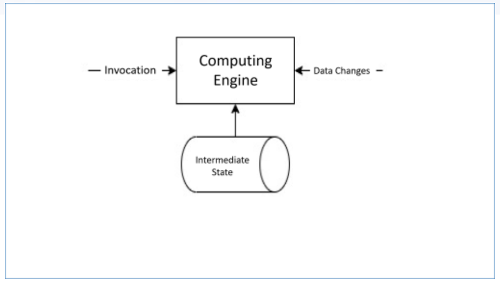
Akulaku Intelligent Risk Control Scenario, Performs Window Feature Computation on 1 Billion Orders, Achieving a Delay Performance of 4 Milliseconds

Difficulties in Feature Computation Process

- **Online Deployment**: Low latency, high timeliness, reflecting data changes as much as possible
- **Offline Analysis**: High throughput
- **Logical Consistency**: The logic for offline analysis and online deployment needs to be completely consistent

OpenMLDB Solution

- Scenario-driven: Business call process driven, real-time calculation results, and on-demand calculation
- **Solution**: 1) Use SQL as a bridge between offline and online; 2) Online time sliding window based on temporal database




```
(product_id:1,  price:2     ,update_time:100L  )
(product_id:1,  price:3     ,update_time:200L  )

(product_id:2,  price:3     ,update_time:1000L  )
(product_id:2,  price:4     ,update_time:1201L  )
(product_id:2,  price:3.5   ,update_time:1100L  )
```

```
SELECT COUNT(*)
FROM w100ms

WINDOW w100ms AS
 (PARTITION BY product_id ORDER BY update_time
 ROWS_RANGE BETWEEN 100ms PRECEDING
AND
CURRENT ROW)
```

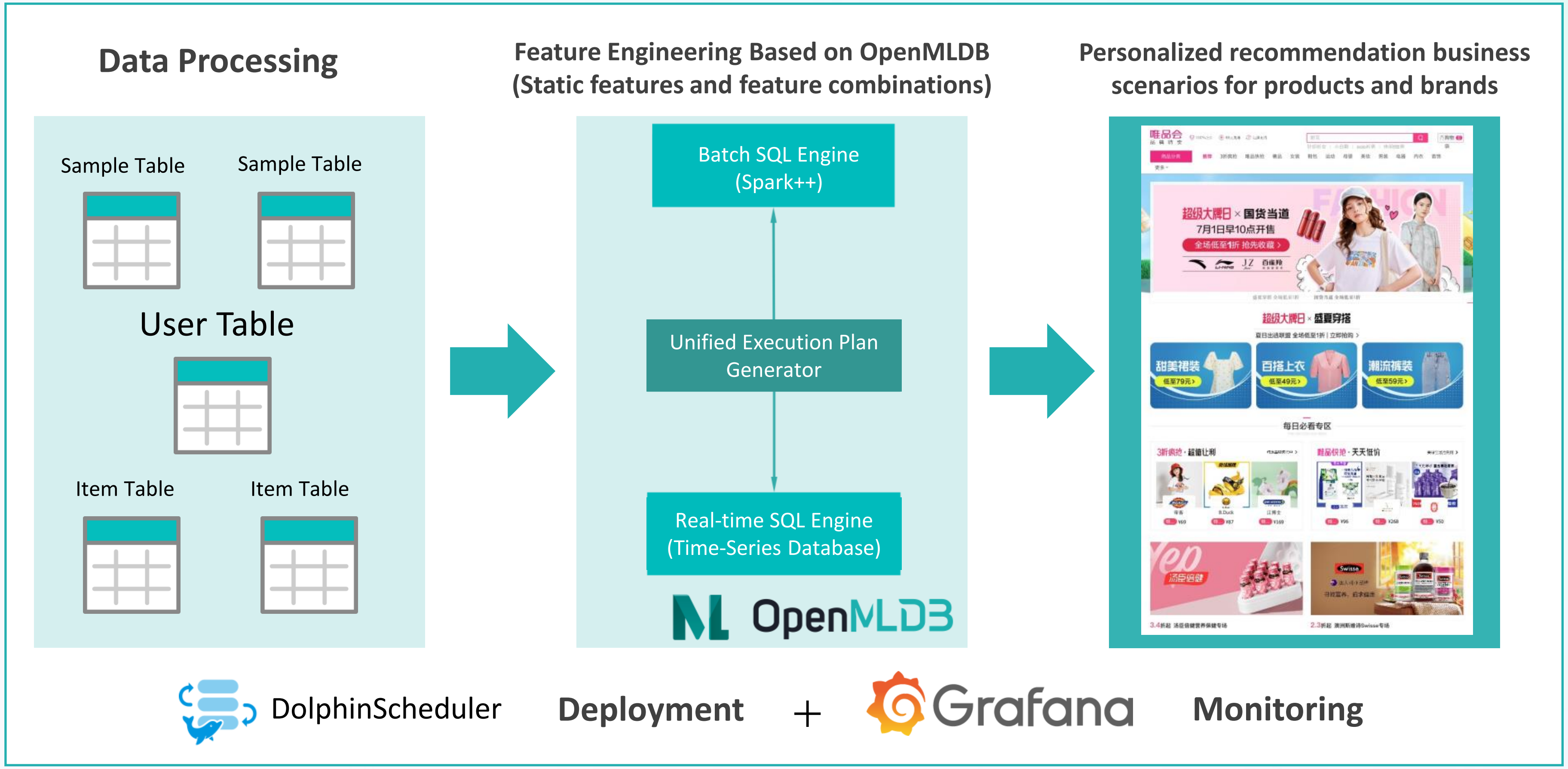
Business Implementation Based on OpenMLDB

- **Scenario**: Real-time computation of the number of orders in the past day
- **Data Volume**: 1 billion order data/day
- **Requirements**: Real-time updates, real-time sliding of time windows, and complex related requirements
- **Test Result**: 4 millisecond delay

akulaku

OpenMLDB

Vipshop Applies OpenMLDB to Personalized Product and Brand Recommendation Scenarios, Resulting in a 60% Improvement in Feature Development Iteration Speed



Feature Development Iteration Speed

5 person-days

↓

2 person-days

Note:

- Sample Table: User behavior table in different scenarios, including exposure, clicking, and bookmarking
- User Table: User profile information for all users
- Item Table: Full quantity information table for different materials

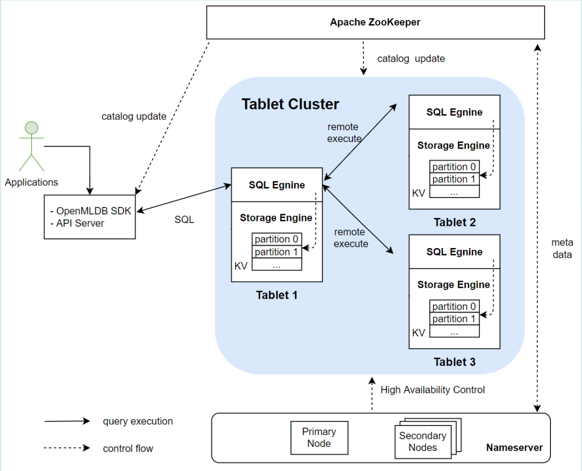
VMALL Adopts OpenMLDB for Real-time Personalized Product Recommendation

Online Real-time Data

- Data Updates in Minutes
- 720 Million Order Data/ day

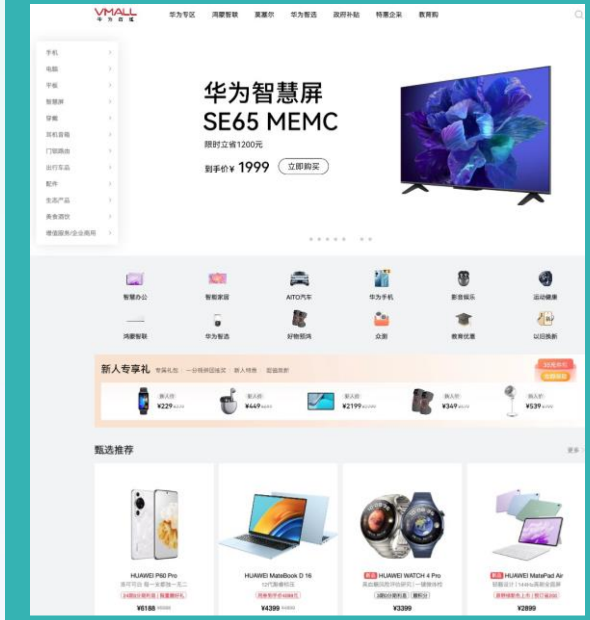
Client Data Table

Real-time Feature Extraction with OpenMLDB



OpenMLDB

Online Feature Deployment in Hours

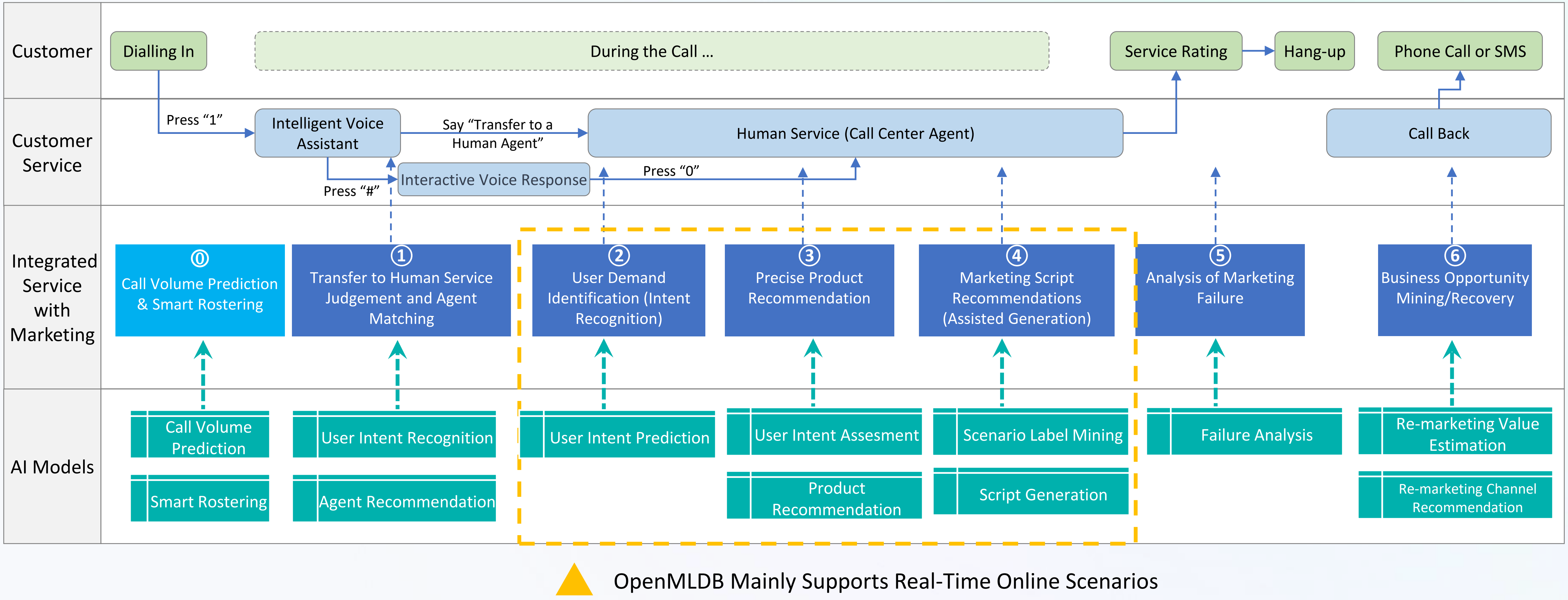


VMALL 华为商城 HUAWEI

OpenMLDB

Leading Telecommunication Operator - Integrated Service with Marketing for Customer Service Hotline Channel

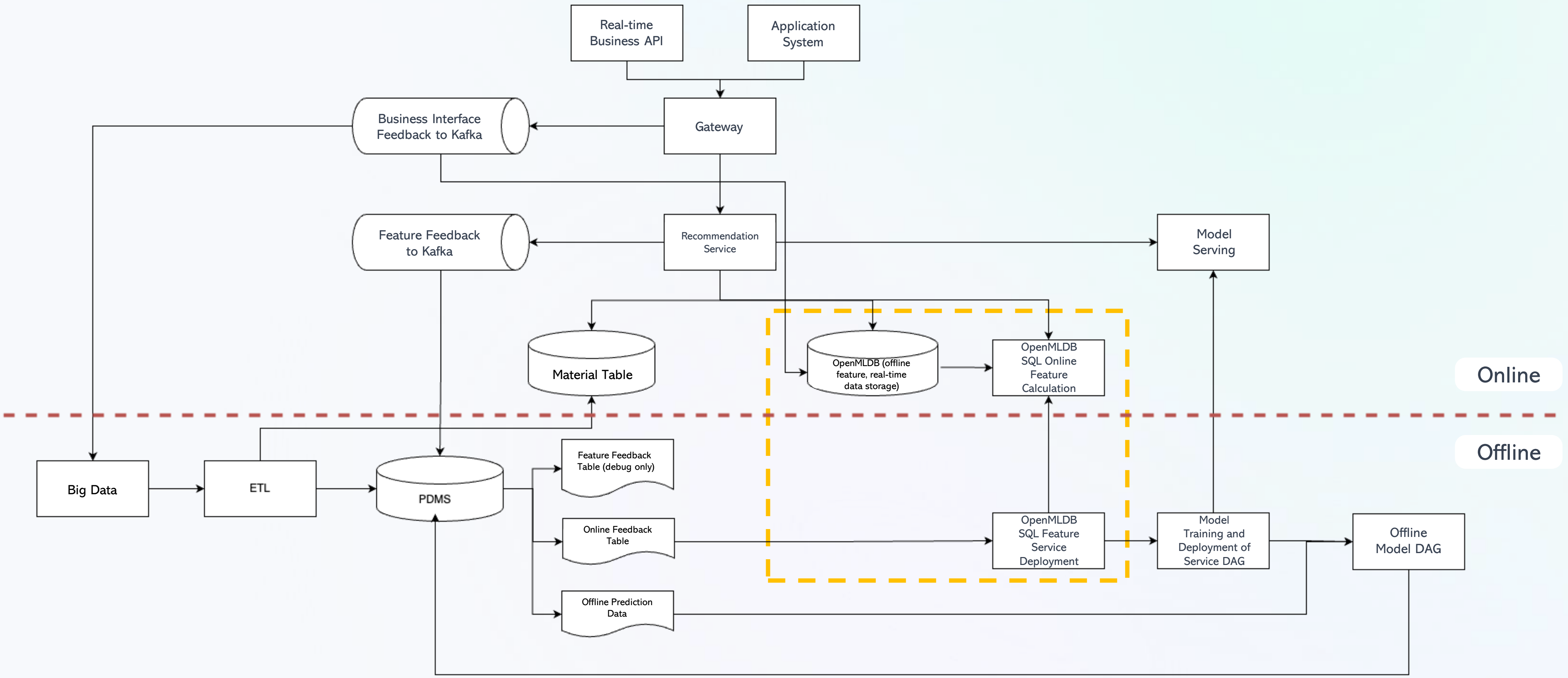
Customer's "journey" through the customer service hotline channel:



OpenMLDB

Leading Telecommunication Operator (Continued) – OpenMLDB Offers Millisecond Level Latency and Guarantees Online and Offline Consistency

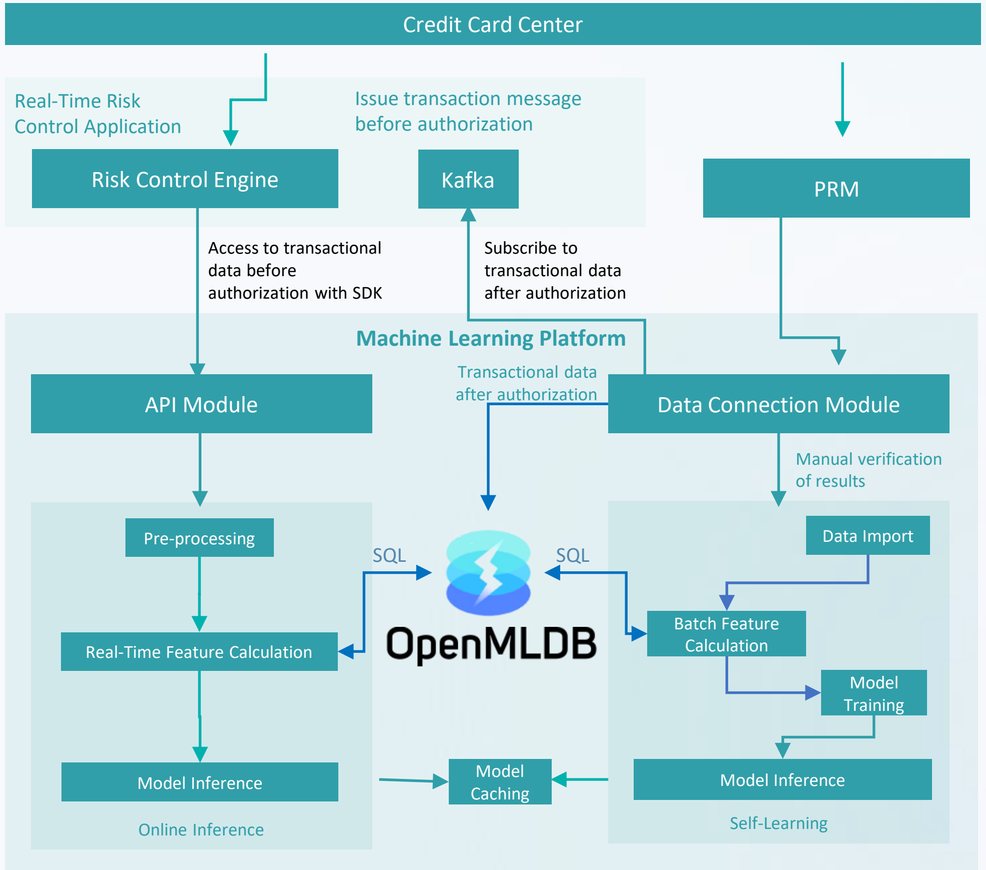
Basic architecture of the integrated service with marketing



OpenMLDB

Leading Bank Builds Real-time Anti-fraud Transaction System with OpenMLDB

Architecture diagram of in-transaction risk control system



Bank requires millisecond-level responses

Customer Requirement: A real-time anti-fraud system with a response time of within 20ms and high recall rate

Solution	Response Time	Admission Rate
Traditional Rule-based	~200ms	Relatively Poor
Client Self-developed System	50ms	Moderate
OpenMLDB-based	<20ms	Excellent

OpenMLDB provides distributed, scalable online prediction services with rapid response capabilities, incorporating millisecond-level changes as features for modeling.

- Integration with real-time stream computing engines and message queues
- High-performance real-time time series data extraction
- Real-time transaction closed-loop data writing back
- Real-time high-concurrency read-write separation support

OpenMLDB

FEBench: First Benchmark for Real-time Feature Computation in Machine Learning

FEBench is the first industrial benchmark model for real-time feature computation in machine learning. It is developed in collaboration with Tsinghua University, the National University of Singapore, and the OpenMLDB community.

- Selected 6 typical scenarios from 118 typical use-case scenarios through modeling as benchmarks
- Compared several common features that support real-time feature computing platforms, including OpenMLDB, Flink, etc. OpenMLDB has significant advantages in handling real-time feature computing scenarios.
- Accepted to the top international database academic conference VLDB 2023 and awarded Best Industry Paper Runner-Up.

Table 4: The statistics of selected datasets.

Cluster	Task	Tables	Columns	Tuples
Q0	*Ride Prediction*	1	11	2.62×10^6
Q1	*Flu Forecast*	1	6	3.54×10^6
Q2	*Energy Forecast*	8	61	8.0×10^6
Q3	*Sales Prediction*	7	85	1.5×10^{10}
Q4	*Loan Evaluation*	9	245	1.0×10^9
Q5	*Fraud Detection*	10	773	1.3×10^{11}

6 typical scenarios from 118 typical scenarios as benchmark for testing

GitHub Open-source：
https://github.com/decis-bench/febench

Welcome to the OpenMLDB community

WeChat Group (Chinese)

Slack

LinkedIn Group

OpenMLDB Official Website https://openmldb.ai/

GitHub: https://github.com/4paradigm/OpenMLDB

Thanks

OpenMLDB Official Website https://openmldb.ai/

GitHub: https://github.com/4paradigm/OpenMLDB