DORIS | Webinar Release

Apache Doris 3.1 版本

Lakehouse 湖仓一体功能增强

9月25日 周四 19:30-20:30

观看直播

Apache Doris 3.1

Lakehouse at New Heights

3.1 Release Webinar

Speaker:

Qi Chen, Apache Doris Committer

DORIS SELECTDB

Agenda

1. Opening: The Road to Unified Lakehouse
2. Async Materialized Views on Data Lakes
3. Iceberg & Paimon Enhancement
4. Performance Improvement
5. Wrap-up & Looking Ahead

Opening:

The Road to Unified Lakehouse

Apache Doris has evolved to become a unified engine
for real-time analytics and data lakes. This seamless integration addresses the traditional challenges of separate data systems while systems while delivering high performance across diverse workloads.

→ Bridging the gap between data warehouses and data lakes

DORIS SELECTDB

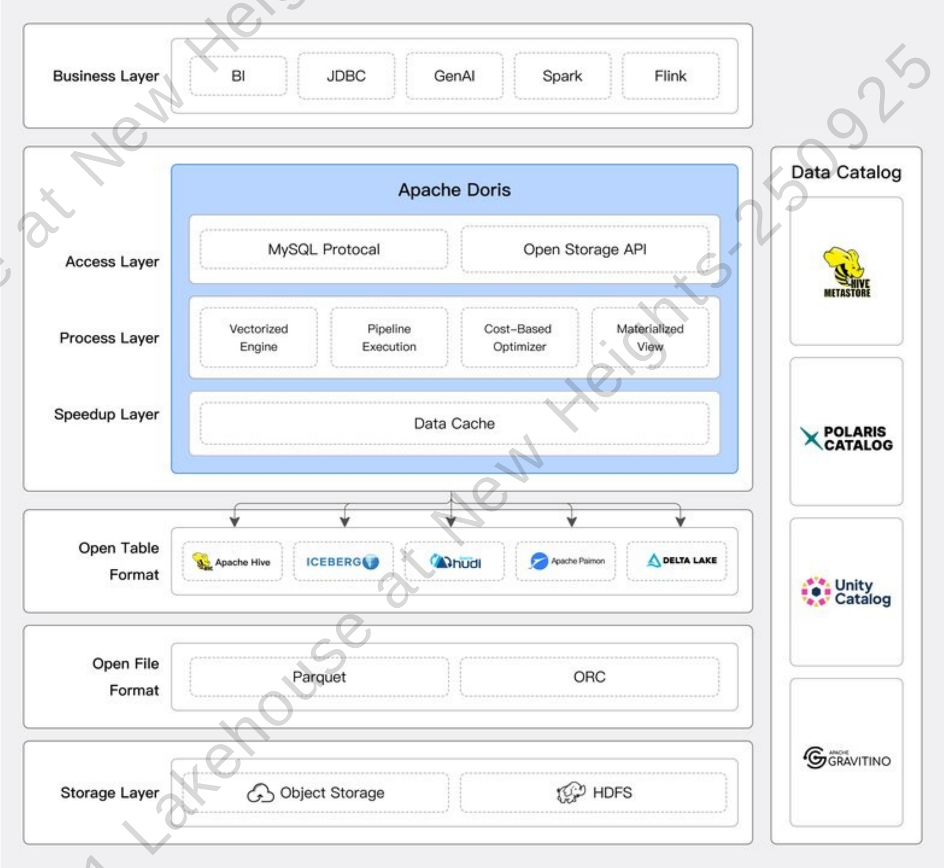
Apache Doris — Unified Analytics Engine

Unified Engine

that seamlessly integrates real-time analytics with data lake capabilities, eliminating complex pipeline architectures

High Performance

Enables sub-second queries across petabytes of data from both real-time and historical sources



Business Layer: BI, JDBC, GenAI, Spark, Flink

Apache Doris

Access Layer: MySQL Protocal, Open Storage API

Process Layer: Vectorized Engine, Pipeline Execution, Cost-Based Optimizer, Materialized View

Speedup Layer: Data Cache

Open Table Format: Apache Hive, ICEBERG, hudi, Apache Paimon, DELTA LAKE

Open File Format: Parquet, ORC

Storage Layer: Object Storage, HDFS

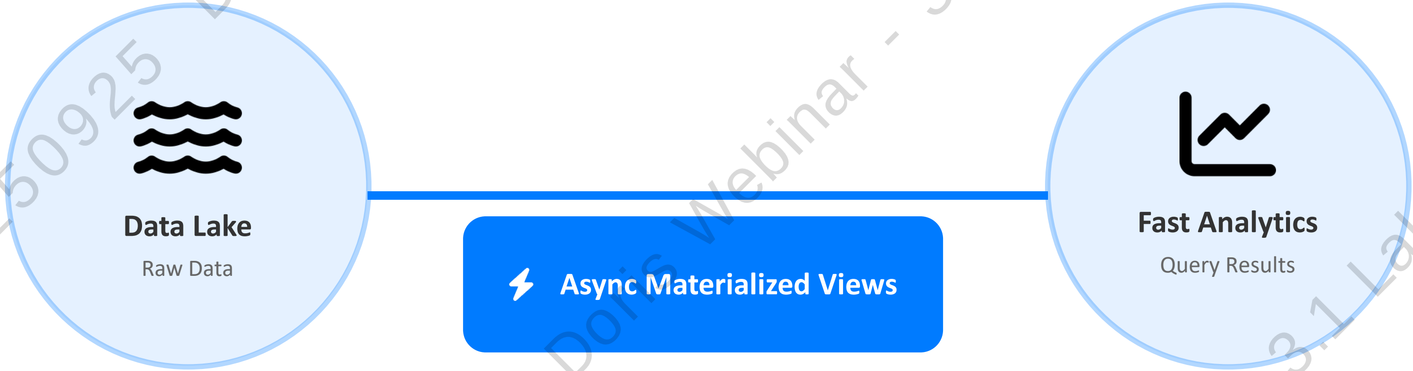
Data Catalog: HIVE METASTORE, POLARIS CATALOG, Unity Catalog, APACHE GRAVITINO

1.

Async Materialized Views on Data Lakes

Bridging the gap between raw data lakes and high performance analytics

The Speed Booster for your lakehouse architecture



Materialized View: The Bridge Between Lake and Database

Materialized views are pre-computed, persistent result sets that transform complex analytical queries into simple table lookups, significantly accelerating query performance while maintaining data freshness.

Materialized Storage:

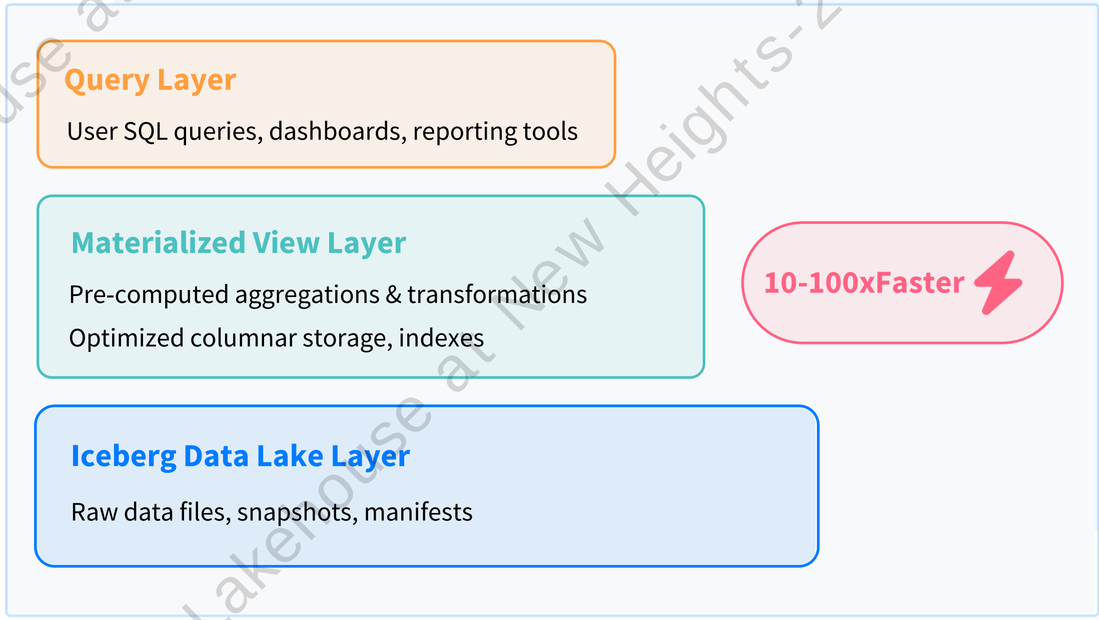
Apache Doris stores pre-computed aggregates, joins, and filtered datasets in its highly optimized columnar format, dramatically reducing query processing time.

Transparent Query Rewrite:

Queries automatically leverage materialized views without requiring any changes to application code, making integration seamless.

Partition Level Refresh:

Instead of rebuilding the entire materialized view when source data changes, Doris intelligently refreshes only affected partitions, dramatically reducing refresh overhead and ensuring data freshness.



Query Layer

User SQL queries, dashboards, reporting tools

Materialized View Layer

Pre-computed aggregations & transformations

Optimized columnar storage, indexes

Iceberg Data Lake Layer

Raw data files, snapshots, manifests

Materialized View: What’s New in 3.1

Apache Doris 3.1.0 extends asynchronous materialized view capabilities to all major data lake formats, enhancing the lakehouse lakehouse experience with partition-aware processing.

Partition Refresh

Efficiently update only changed partitions instead of rebuilding entire rebuilding entire materialized views

Partition-Level Transparent Rewrite

Automatically compensate for data freshness during query execution execution

Key Benefit:

Query acceleration on data lakes with minimal maintenance overhead overhead

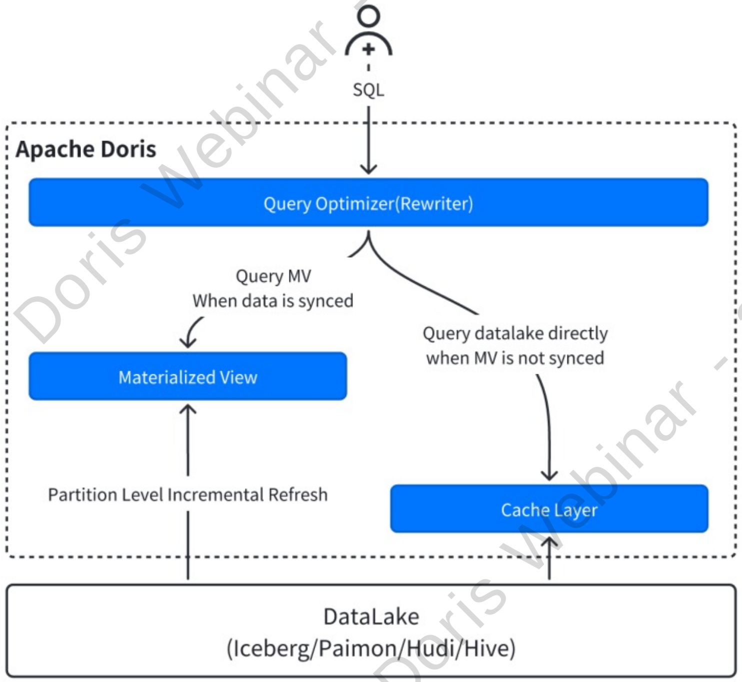
Lake Table	Partition Refresh	Partition-Level Transparent Rewrite
Hive	Supported	Supported
Iceberg	Supported	Supported
Paimon	Supported	Supported
Hudi	Supported	Supported(*)

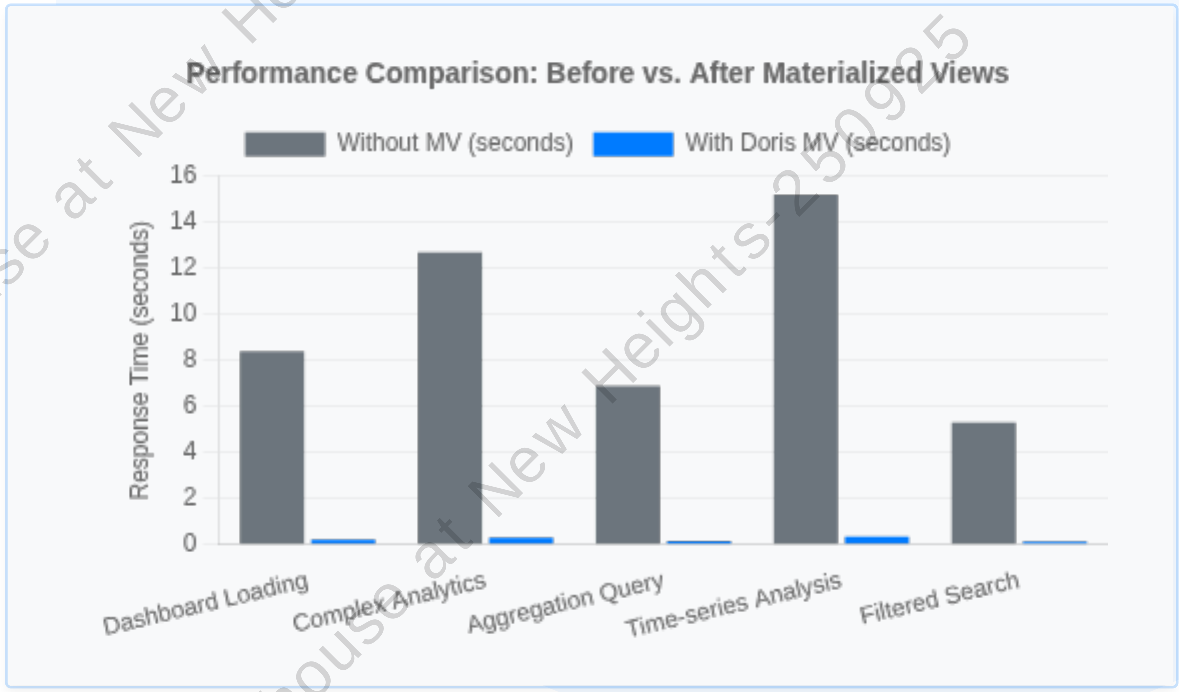
** Cannot identify if partitions are synchronized; suitable for manually refreshing specified partitions*

DORIS SELECTDB

Materialized View: Showcase

Real-world case study: Large e-commerce analytics platform with 50TB+ of customer behavior data in Iceberg tables, experiencing performance issues with complex analytical queries across multiple dimensions.





42×
Average query speedup

98%
Reduction in scan size

23 ms
Avg. dashboard response

5x
Concurrent user capacity

2.

Iceberg & Paimon Enhancement

Modern data lakes are evolving beyond simple file collections into sophisticated table formats with rich metadata. Apache Doris 3.1.0 Apache Doris 3.1.0 delivers enhanced support for leading open table formats.

- Git-style versioning with Branch & Tag support
- System tables for metadata transparency

+

Branch & Tag: Git-Style Data Control

Create branches

for development and testing environments without affecting production data, enabling safe experimentation

Apply tags

to create snapshots for audit, compliance, or recovery points, preserving data states for long-term reference

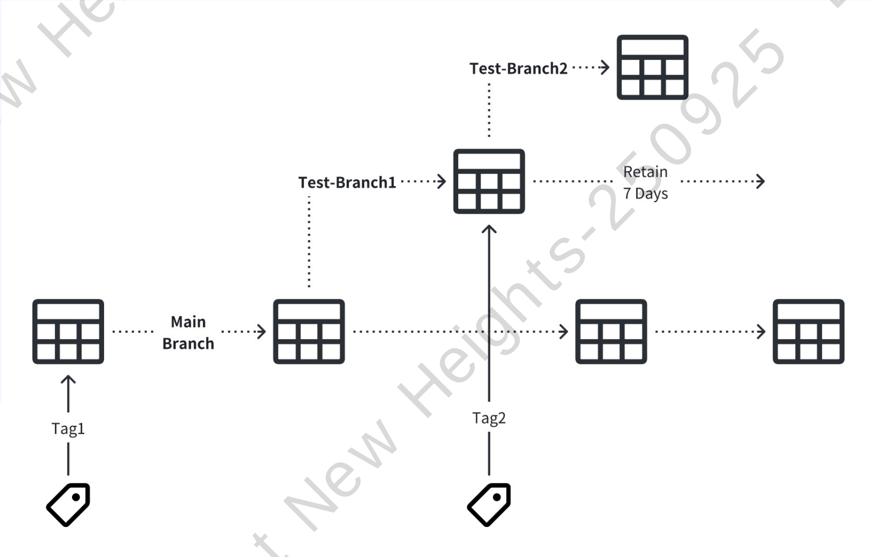
Easy Operation

Perform Git-like operations directly on your lakehouse tables through familiar SQL syntax

```sql
-- Create a branch in Iceberg
ALTER TABLE iceberg_tbl CREATE BRANCH b1;

-- Insert data into specific branch
INSERT INTO iceberg_tbl@branch(b1) VALUES (...);

-- Read from Paimon tag
SELECT * FROM paimon_tbl@tag(release_v1);
```



Business Benefits:

Reduce risk while enabling fearless iteration on data. Create isolated testing environments, perform what-if analysis, and guarantee regulatory compliance with immutable snapshots.

System Tables:

Opening the Metadata Black Box

- Direct SQL access to Iceberg and Paimon system tables for snapshots, snapshots, partitions, and file layout details

- Easily inspect metadata structures with simple SQL queries, no specialized tools required

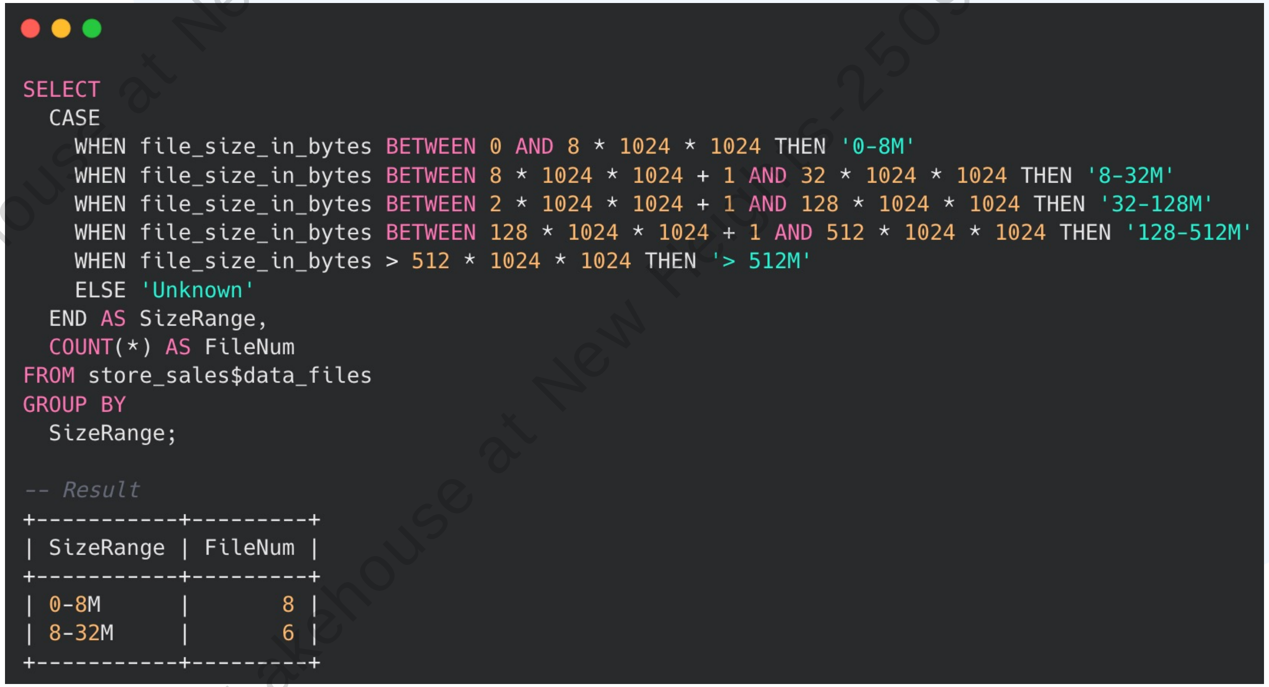
- Simplify lake governance, debugging, and performance tuning tuning through transparent metadata visibility

```
SELECT * FROM iceberg_table$snapshots;

SELECT * FROM iceberg_table$manifests;

SELECT * FROM paimon_table$files
```

> No more black boxes:
>
> Turn metadata into an accessible resource for developers and and administrators



```
SELECT
  CASE
    WHEN file_size_in_bytes BETWEEN 0 AND 8 * 1024 * 1024 THEN '0-8M'
    WHEN file_size_in_bytes BETWEEN 8 * 1024 * 1024 + 1 AND 32 * 1024 * 1024 THEN '8-32M'
    WHEN file_size_in_bytes BETWEEN 2 * 1024 * 1024 + 1 AND 128 * 1024 * 1024 THEN '32-128M'
    WHEN file_size_in_bytes BETWEEN 128 * 1024 * 1024 + 1 AND 512 * 1024 * 1024 THEN '128-512M'
    WHEN file_size_in_bytes > 512 * 1024 * 1024 THEN '> 512M'
    ELSE 'Unknown'
  END AS SizeRange,
  COUNT(*) AS FileNum
FROM store_sales$data_files
GROUP BY
  SizeRange;

-- Result
+-----------+---------+
| SizeRange | FileNum |
+-----------+---------+
| 0-8M      |       8 |
| 8-32M     |       6 |
+-----------+---------+
```

Display data file size distribution. This can help identify if there are too many

Enhanced Data Integration Capabilities

Iceberg & Paimon Rest Catalog Enhancements

- Vended Credential Support for secure cloud-native authentication authentication
- Mainstream Catalog Integration with AWS Glue, Polaris, Alibaba DLF Alibaba DLF and Gravitino catalogs

Hadoop Ecosystem Integration

- Multi-Kerberos Environment Support for secure access to different different Hadoop clusters
- Multi-Hadoop Configuration enabling connection to diverse Hadoop deployments with different versions and configurations

Unity Catalog | POLARIS CATALOG | APACHE GRAVITINO | DLF | hadoop HDFS | HIVE METASTORE

> These enhanced data integration capabilities establish Apache Doris as a true unified data analytics platform that seamlessly connects to diverse data sources across cloud and on-premises environments, providing a single interface for analyzing data wherever it resides.

Why It Matters to You

For Developers

Safely experiment with Git-like branches for data, enabling fearless iteration and innovation without affecting production environments. Create dedicated development branches, test new transformations, and merge only when ready. The result: faster development cycles, reduced time-to-insight, and freedom to experiment without risk.

- ✓ Safer sandboxing for data experiments
- ✓ Rapid iteration cycles
- ✓ No-risk innovation environment

For Operators

Gain unprecedented transparency into table metadata, file layouts, and query execution. Debug complex issues by directly querying system tables with familiar SQL—no need for specialized tools or external systems. Monitor data quality, optimize performance, and troubleshoot problems with powerful, accessible insights.

- ✓ Full metadata transparency
- ✓ Easier debugging with SQL
- ✓ Self-service optimization

For Enterprises

Meet regulatory requirements with auditable data history and immutable versioning. Create snapshot tags for compliance checkpoints, maintain full data lineage, and enable point-in-time recovery for disaster scenarios. Enhance governance with complete visibility into data access patterns and schema evolution over time.

- ✓ Reliable compliance & audit trails
- ✓ Immutable versioning for recovery
- ✓ Enhanced data governance

Iceberg & Paimon Feature Support

Iceberg – Currently Supported

- Catalog Integration: HMS, Glue, Rest(Unity, Polaris, Gravitino, …)
- Basic Query with Iceberg V1/V2 table format
- Native Position/Equality Delete Read
- Append Only Insert/ Insert Overwrite
- Time Travel & View
- Create/Drop Database and Table
- Alter Table Schema
- Create/Drop/Read/Write Branch and Tag
- System Table

Iceberg - In Progress

- Update/Delete & V3 table format
- Variant Data Type
- Data Rewrite & Snapshot Management

Paimon - Currently Supported

- Catalog Integration: HMS, FileSystem, Rest(DLF)
- Basic Query
- Native Deletion Vector Read Support
- Time Travel
- Batch Incremental Query
- Read Branch and Tag
- System Table

Paimon - Planned

- Write Support
- Create/Drop Database and Table
- Alter Table Schema

3.

Performance Improvement

Smarter Pruning, Faster Execution

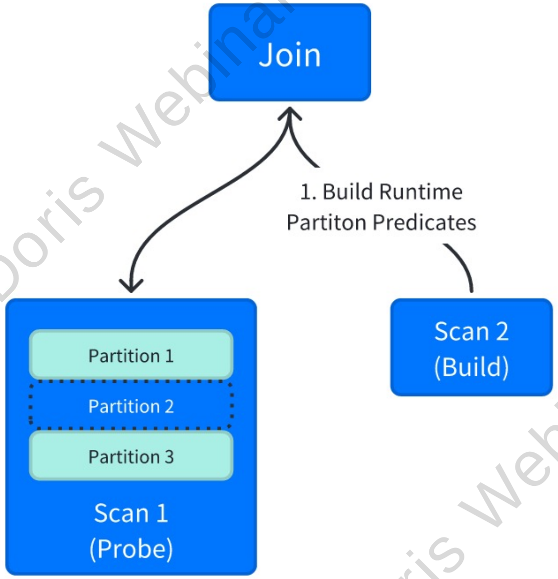
Apache Doris 3.1.0 brings significant performance enhancements to data lake queries through intelligent optimization techniques. These improvements deliver faster query responses while dramatically reducing resource consumption.

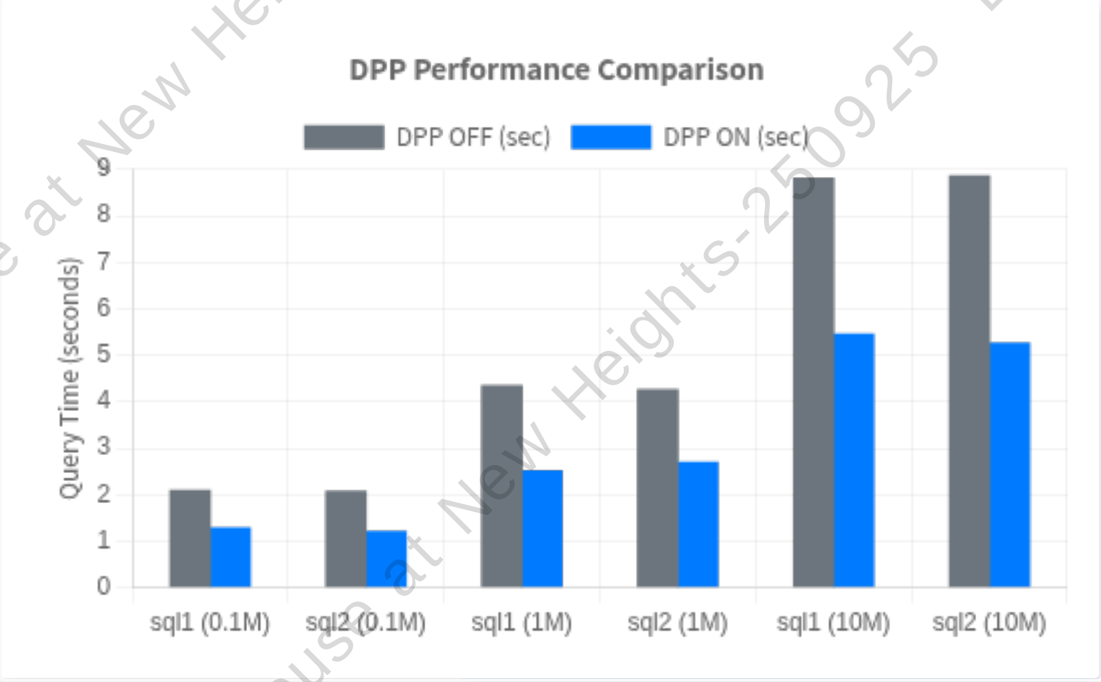
- Up to 40% faster queries with dynamic partition pruning
- Reduced FE memory footprint with batch fragment execution

DORIS SELECTDB

Dynamic Partition Pruning

Dynamic partition pruning (DPP)
(DPP)
intelligently skips irrelevant partitions at runtime by generating partition predicates from join operations, significantly reducing I/O and accelerating queries



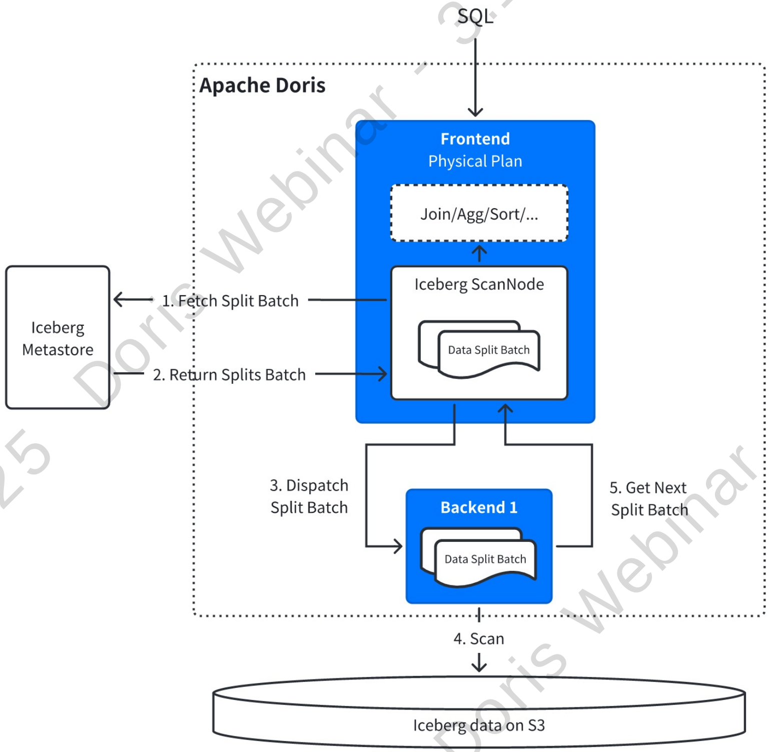


40% Average performance improvement across all test cases, showing consistent benefits regardless of query type or data volume

DORIS SELECTDB

Split Batch Mode

Split Batch Mode optimizes query execution by restructuring how Frontend and Backend nodes interact, significantly improving resource utilization and query efficiency.



Key Benefits

Reduced Frontend Memory Usage

Frontend only maintains partial data shard information throughout the query process, process, significantly reducing memory pressure during large queries

Lower RPC Overhead

Each RPC between Frontend and Backend only processes a single batch of shard information, reducing timeout risks and network bandwidth pressure

Improved Pipeline Efficiency

Through multi-level pipeline architecture, Frontend can acquire shard information while Backend simultaneously processes data, maximizing resource utilization

Before: Wait for all shards to be collected before execution

After: Progressive batch processing with parallel execution

4.

Apache Doris Looking Ahead

Real-time + AI Integration

Deeper integration between streaming real-time data and AI-powered analytics

Advanced Lakehouse Federation

Seamless cross-catalog querying and optimization, bringing true multi-lake query capabilities

Unified Lakehouse Management

Making open lake table formats first-class citizens, offering the same seamless experience as databases.

Unified Lakehouse Philosophy

Breaking down silos between data lakes and database while providing a consistent, high-performance query layer for all your data

DORIS | Webinar Release

回放与演讲资料获取

请关注 SelectDB 公众号发送 **20250925**

联系我们

🌐 www.selectdb.com

☎ 400-092-6099

微信公众号

免费试用

在线咨询

加入社区