Competitive Battle Card: ClickHouse

Generated: 10/25/2026, 10:55:56 AM | Company: PhoenixAI | Target Competitor: ClickHouse | Target Job Title: Data Engineer | Specified Industry: Unspecified

Competitor and Product

Competitor Name:
ClickHouse

Value Proposition:
Fast open-source OLAP database for real-time analytics, delivering sub-second queries on billions of rows with columnar storage and vectorized execution.

Primary Products:
ClickHouse Cloud (fully managed), ClickHouse (open-source self-hosted)

Our Key Features

Distributed MPP joins on normalized data without precomputation
Runs complex multi-table joins on the fly against normalized schemas with sub-second latency. No need to pre-flatten or denormalize—agents and dashboards get the join they asked for in one query.

Sub-second freshness with streaming upserts and deletes
Ingests streaming data from Kafka and Flink CDC at 100K+ events/sec with updates queryable in under 5 seconds. Primary key tables support real-time upserts and deletes without the FINAL tax or query latency degradation.

Unified SQL across real-time and lakehouse data
Query streaming tables and lakehouse data (Iceberg, Delta, Hudi) in place with a single SQL statement. Years of historical data join with live tables—no pipeline to build, no data to copy.

Isolated warehouses for workload isolation
Agent queries, BI dashboards, and ingestion each run in their own compute warehouse on the same data. Agent bursts don't contend with production dashboards, and each workload scales independently.

Cost-based optimizer for unpredictable AI-generated SQL
Automatically handles novel joins and LLM-generated SQL without manual tuning. Global runtime filters and query feedback deliver 10x–100x speedups on multi-join patterns, even when statistics are incomplete.

ClickHouse Key Features

Columnar Storage and Vectorized Query Execution
Stores data by column rather than row, reading only needed columns to reduce I/O. Vectorized execution processes data in batches using CPU cache and SIMD instructions, delivering blazing-fast analytical query performance at scale.

Real-Time Streaming Ingestion from Kafka
Native Kafka table engine and ClickPipes managed ingestion support millions of events per second from Kafka, Confluent Cloud, and Amazon MSK, with data queryable within seconds of arrival for real-time dashboards and monitoring.

Lakehouse Analytics with Iceberg and Delta Lake
Reads and writes over 70 data formats including Iceberg and Delta Lake directly in place on cloud object storage, combining data lake flexibility with database-grade ACID transactions and fast analytical queries.

Auto-Scaling and Managed Operations (Cloud)
ClickHouse Cloud provides serverless compute that auto-scales and auto-suspends based on workload, with automatic backups, upgrades, high availability, and monitoring dashboards eliminating cluster management overhead.

High Compression and Cost Efficiency
Advanced compression algorithms (LZ4, ZSTD, Delta, Gorilla) achieve 10x better storage efficiency than traditional databases, significantly reducing storage costs while maintaining fast decompression for query performance.

Top Prospect Initiatives

A. Build and maintain scalable real-time data pipelines from streaming sources (Kafka, Flink CDC) with sub-second data freshness

B. Design lakehouse architectures that unify streaming and historical data (Iceberg, Delta) for analytics and ML workloads

C. Support AI agent and ML workloads with low-latency query infrastructure that handles unpredictable, complex SQL patterns

D. Optimize query performance and reduce infrastructure costs through right-sized compute, partitioning, and workload isolation

E. Implement data governance, quality checks, schema management, and role-based access controls across pipelines

F. Reduce pipeline complexity by eliminating redundant ETL and enabling direct queries on lakehouse tables

G. Ensure high concurrency and sub-second latency for customer-facing analytics and embedded dashboards

H. Monitor pipeline health, resolve failures, and implement observability across streaming and batch workloads

I. Collaborate with data scientists, analysts, and engineering teams to translate requirements into reliable data infrastructure

J. Evaluate and adopt cloud-native data platforms that scale elastically and integrate with existing tools

Where We Win

Streaming upserts queryable in seconds
Ties to Initiative A

Primary key tables handle 100K+ events/sec with upserts and deletes queryable in under 5 seconds—no FINAL tax or latency degradation. Intuit achieves 2s total latency for real-time ML predictions.

Unified SQL on streaming and lakehouse
Ties to Initiative B

Query Iceberg, Delta, and Hudi in place alongside streaming tables in one SQL statement. 8.2× faster than Trino on Iceberg in TPC-DS 1TB benchmark. No data copy, no duplicate pipeline to maintain.

Cost-based optimizer for AI-generated…
Ties to Initiative C

Automatically handles novel joins and LLM-generated SQL without manual tuning. Demandbase replaced ClickHouse architecture that couldn't handle unpredictable AI agent queries; optimizer adapts at runtime.

Isolated warehouses for workload…
Ties to Initiative D

Agent, BI, and ingestion workloads each run in their own compute on the same data. Demandbase achieved 60% hardware reduction and 49!'1 cluster consolidation after replacing ClickHouse sprawl.

Distributed MPP joins without denorm
Ties to Initiative G

Complex multi-table joins on normalized schemas with sub-second latency. Coinbase queries 573B rows across 300+ tables at <1s latency. No pre-flattening or pre-aggregation required.

Competitor Considerations

Vectorized columnar query execution
Ties to Initiative G
Where We Compete

PhoenixAI also uses vectorized execution, but adds distributed MPP joins. Coinbase's TPC-H 1TB test: ClickHouse failed 12 of 22 queries with OOM errors; PhoenixAI completed all with better P50–P99 latency.

Native Kafka streaming ingestion
Ties to Initiative A
Where We Compete

PhoenixAI also ingests Kafka at 100K+ events/sec, but commits upserts in place with sub-10-second freshness. ClickHouse's merge-on-read forces a choice: stale rows or the FINAL tax that degrades query speed.

Iceberg and Delta Lake support
Ties to Initiative B
Where We Compete

PhoenixAI also queries lakehouse formats in place, but unifies them with streaming tables in one SQL statement. PhoenixAI delivers 8.2× the performance of Trino and 1.3× Databricks on Iceberg in TPC-DS 1TB.

High compression and storage efficiency
Ties to Initiative D

Advanced compression (LZ4, ZSTD, Delta, Gorilla) achieves 10x better storage efficiency than traditional databases, significantly reducing storage costs while maintaining fast decompression.

Auto-scaling serverless compute

ClickHouse Cloud auto-scales and auto-suspends based on workload with managed backups and upgrades, eliminating cluster management overhead for teams prioritizing operational simplicity.

Customer References

- Demandbase — Case Study
- Coinbase — Case Study
- Conductor — Case Study
- Eightfold.ai — Case Study
- Pinterest — Case Study

Pricing Overview

Usage-based pricing at $0.0065 per unit (1% of standard CCU) for PhoenixAI Cloud; $175/month per CPU core for PhoenixAI Enterprise. Flexible plans with no end date, cancel anytime.

Key Talking Points

Q: This product is too expensive

A: PhoenixAI at $0.0065 per unit often costs less than ClickHouse Cloud's $0.22–$0.39 per unit-hour. Demandbase cut hardware 60% and storage 90% replacing ClickHouse, consolidating 49 clusters to 1. You pay for what you use, and workload isolation means you don't over-provision for burst traffic.

Q: How are you any better than ClickHouse?

A: ClickHouse runs JOINs on a single node and forces a FINAL tax for fresh data; PhoenixAI runs distributed MPP joins sub-second with upserts committed in-place at 100K+ events/sec. Our CBO handles LLM-generated SQL ClickHouse can't.

Q: We already have a similar solution, why would we replace it with you?

A: If you're struggling with unpredictable AI agent queries, real-time upserts that stay fresh without latency hits, or complex joins that require denormalization, PhoenixAI solves those.

Q: Can you really handle streaming and lakehouse data in one query?

Key Talking Points (Cont.)

A: Yes. PhoenixAI runs unified SQL across Kafka streams, Iceberg, Delta, and Hudi in place—no copy, no pipeline. Coinbase queries 573B rows across live Kafka feeds and lakehouse tables with <1s latency.

Q: How do you prevent AI agent bursts from breaking our dashboards?

A: Isolated warehouses. Agent queries run in their own compute; BI dashboards run in theirs, both on the same data. Demandbase's agents hammer petabytes of Iceberg tables while customer dashboards keep second-level SLAs. Each workload scales independently—no contention, no manual throttling required.