Scaling Cyber Analytics: The ClickHouse Advantage

A deep dive into architectural evolution and performance optimization for our next-generation data platform.

Ajit Bhat - Engineering Manager, NetScout Systems

Afzal Khan - Principal Software Engineer, NetScout Systems

Agenda: Unpacking Our Data Transformation

01

Our Analytical Foundation

A brief look at the evolution of our data platforms and the challenges we face with escalating volumes.

02

The ClickHouse Initiative

Introducing our strategic move to ClickHouse for enhanced performance and scalability.

03

Design & Implementation Deep Dive

Exploring schema, indexing, and replication strategies for optimal deployment.

04

Performance Benchmarks

Examining ingestion and query rates, and the impact on our cybersecurity analytics.

05

Next Steps & Future Outlook

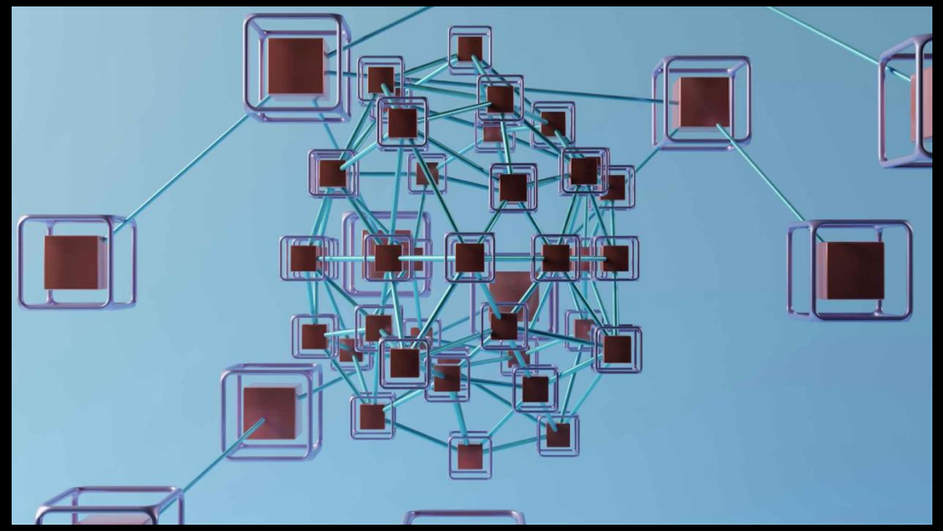
Roadmap for integration, system tuning, and maximizing ClickHouse capabilities.

Tracing Our Data Platform Evolution

From our origins in network monitoring to advanced cybersecurity analytics, our data infrastructure has continuously adapted to meet escalating demands.

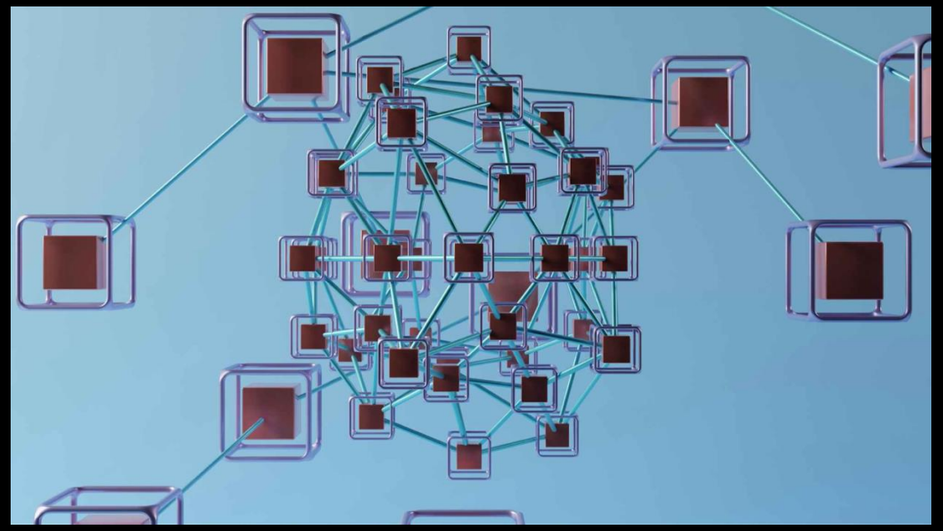
Network Monitoring to Advanced Analytics

Initially focused on deep packet inspection and network performance, our systems have evolved to handle complex behavioral analytics crucial for threat detection.

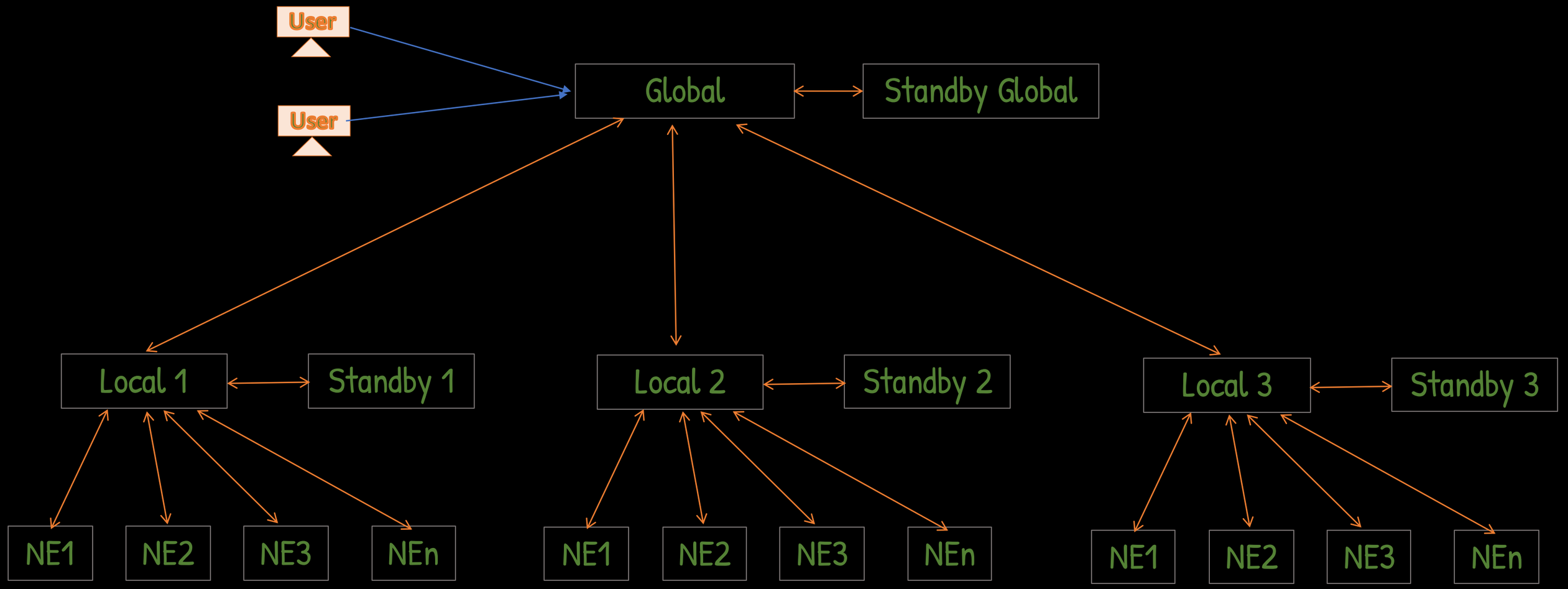


Database Journey

We've transitioned through several database architectures, from traditional relational systems to highly scalable distributed data stores, each iteration pushing the boundaries of performance.



Product Deployment



Legacy Data Stack

- Home-grown columnar database
- PostgreSQL (security events)
- Custom index framework for search
- Write statistics:
  - 100M network records per 5 min (Columnar DB)
  - 100K security events/ 5 min (PostgreSQL)
- Internal replication to redundant servers

The Scale of Data

Our data volumes have grown exponentially, posing significant challenges for traditional database architectures and demanding innovative solutions for storage and query efficiency.

For a physical server appliance (256G memory and 92 CPUs):

100M

Records per 5 min

From 500K to 3M, now processing up to 100M new records every five minutes.

1M/s

Peak Ingestion Rate

Sustained ingestion rates can reach up to 1 million events per second during business hours.

450B

Total Raw Data Rows

Our raw data archive now exceeds 450 billion rows, underpinning all our analytics.

200B

Rollups & Warehouse Rows

Aggregated and warehoused data contribute an additional 200 billion rows, optimizing common queries.

Challenges with Legacy Database

- **Query Latency** – Enterprise queries too slow, poor support for large-scale analytics
- **String Data type Limitations** – Even migration to PostgreSQL could not solve issues
- **User Experience** – Poor responsiveness for users (analysts and operators)
- **Database Sizing** – Excessive disk requirements as data grew

Why ClickHouse

Continue with what we have working:

- Columnar storage ⟶ faster OLAP queries, ideal for analytics
- Compression ⟶ reduced storage footprint & cost savings
- Real-time ingestion ⟶ supports streaming, not just batch
- Horizontal scaling ⟶ simple to shard & replicate

Revolutionize what we lacked:

- **Query Response / user experience!**

Schema & Table Design

Schema Design

- ORDER BY aligned with query filters
- Denormalization to reduce joins

Table Engines used

- ReplicatedMergeTree → redundancy & HA
- ReplicatedReplacingMergeTree → cached/cleaned data
- Distributed → automatic query fan-out & result merge across shards

Partitions & Rollup

- Daily partitions for efficient Purging
- Rollup tables for hourly/daily summaries

Function & Cluster

- UDFs: utilization, bitrate, packet rate
- Cluster with sharding + replicas

Data Ingestion

CSV Imports

- Chunked CSV files
- Split for higher throughput

Streaming

- Real-time ingestion pipeline

Write User Profiles

- Dedicated write threads
- Batch size - Controls how many records are written per operation.
- Write timeout - Limits duration to avoid long-running writes

Roll Ups & Background processing

Roll ups

- Materialized view for 5 minute rollups
- Triggered based 5 min rollups(no Union support in MV) and time based Hourly & daily rollups

Analytics Roll Ups

- Structured rollups for analysis

Roll Up User Profiles

- Dedicated query threads
- Query timeouts
- Write timeouts
- batch Sizes

Query Performance & Reads

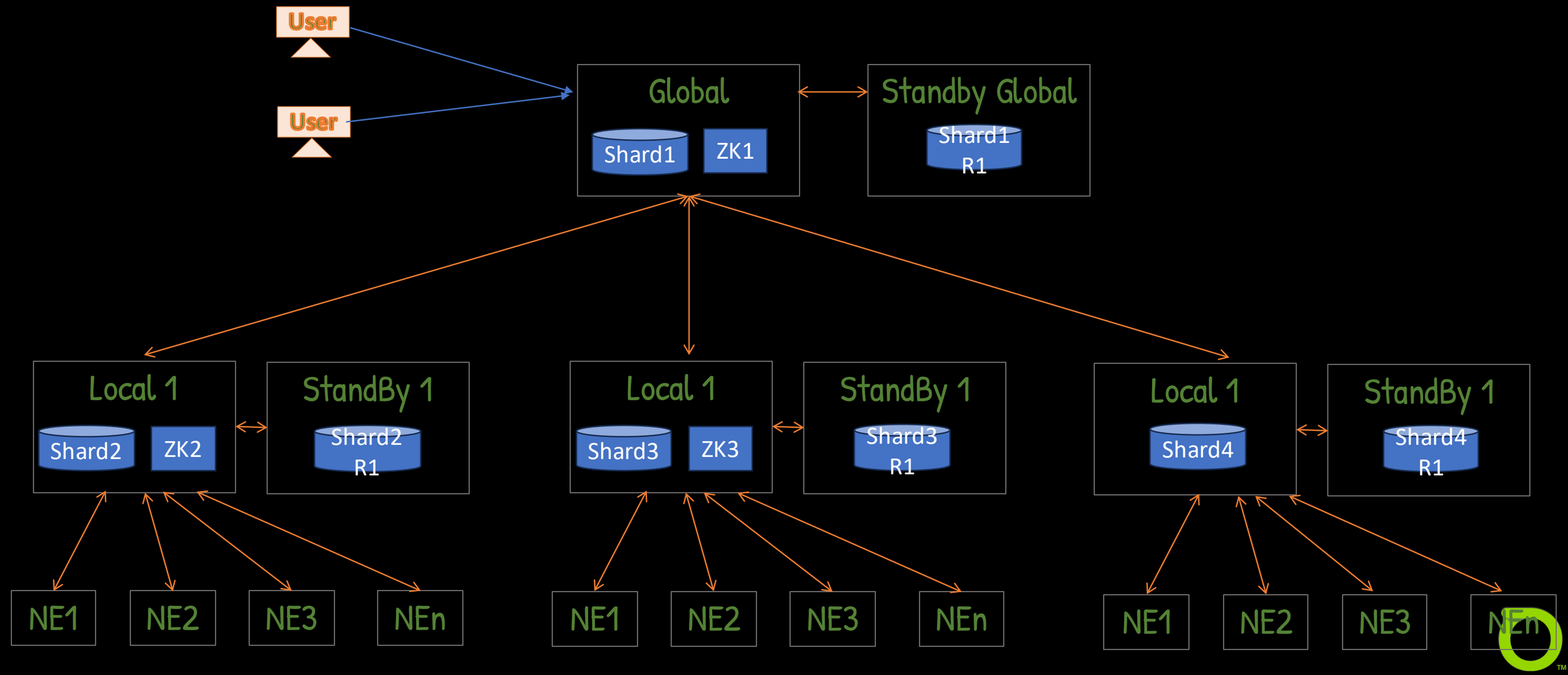
Extraction Layer

- Framework + Plugin-based structure
  - Easy integration of new data sources (e.g., different databases, APIs).
  - Independent development and deployment of plugins without impacting the core.
  - Better code maintainability and reusability across different extraction pipelines.
- This architecture enables:
  - Independent query-level tuning and testing.
  - Clean separation of logic for better maintainability and debugging.
  - Flexibility to inject optimizations based on evolving data patterns.

Read User Profiles

- Query execution characteristics (e.g. batch size, timeout, isolation).
- Performance tuning parameters tailored for different use cases (e.g. real-time vs. batch reads).
- Helps isolate and fine-tune read-intensive workloads without impacting other operations

Product Deployment with ClickHouse



Key Takeaways And Concerns:

Key Takeaways:

- **High Query Performance:** ClickHouse processes hundreds of millions to over a billion rows per second, significantly increasing user query speeds and improved user experience 4x-5x times.
- **Improved Scalability:** Leveraged sharding to eliminate manual query distribution challenges in the legacy system,
- **High Availability:** achieved high availability (HA) through replication with automated failover.
- **Excellent Data Compression:** Achieves up to 40% better compression, reducing storage requirements.

Concerns:

- Zookeeper Management Overhead: Relies on Zookeeper for cluster coordination, which can consume high CPU/memory and require extra maintenance; Will switching to ClickHouse keeper help?.
- Learning Curve for Support: While the community is active, complex issues (e.g., cluster setup or Zookeeper troubleshooting) may require seeking specialized help or expertise.
- Observed system performance degradation during background activities; seeking benchmarks or tuning recommendations to optimize resource usage and avoid impacting real-time workloads.
-

Questions And Answers