



## THE CHAOSSEARCH DATA LAKE PLATFORM

# Elasticsearch Replacement for Log Analytics at Scale

### CHALLENGE

#### Elasticsearch Cannot Scale to Meet Today's Log Management Challenge

While the Elasticsearch stack is free open-source software and is easy to download and install, today's exponential growth in log volume has exposed Elasticsearch's deficiencies for managing logs at scale. Whether deployed on-premises or as a cloud service, an Elasticsearch cluster requires actively managing the underlying compute and storage infrastructure to meet log volume growth and growing demand from end users for more log data and faster queries on dynamic data sets.

Whether cloud infrastructure logs, container logs, security telemetry data, or network device logs, Elasticsearch is bending and breaking under the weight of non-stop log volume growth. CloudOps, DevOps, SecOps, and business users will only demand better access to more logs for longer periods of time which will continue exposing the shortcomings of Elasticsearch at scale:

#### Management Complexity Consumes Resources

An Elasticsearch cluster—whether managed Elasticsearch in the cloud or on-premises—becomes brittle at scale and requires active, hands-on administration. Compute and storage are tightly coupled resulting in cost spikes or failures when log volumes rise (which they always do).

#### Performance and Time to Insights Suffers at Scale

An Elasticsearch cluster requires active management to spin up and spin down servers to accommodate high and query volumes. At scale, an Elasticsearch cluster can cause queries to take hours to complete or they time out, frustrating end users and causing blind spots in analysis.

#### TCO Increases Dramatically at Scale

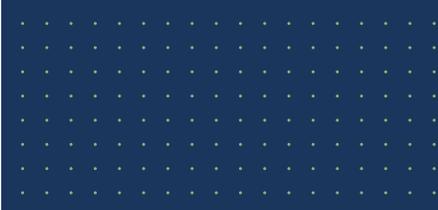
The true cost of an Elasticsearch cluster must include the cost of administration and maintenance. And the inefficient indexing technology of the Lucene database can swell data size. Admins are often forced to make trade-offs—either ingest fewer logs or shorten retention - due to the high cost and unreliability of storing log data at scale.

#### Data Movement and Transformation Add Complexity

With Elasticsearch, you need to push data and transform in Logstash or Fluentd. You also have to continually monitor Elasticsearch exceptions. Plus, you need to build resilient pipelines, configure the stack to ingest and parse logs, ensure log data consistency, reindex outdated indices, and much more.

### A BETTER APPROACH TO LOG ANALYTICS

Imagine sending all your data to your cloud environment in its native format—no parsing or schema changes. ChaosSearch indexes all data as-is, without transformation, while auto-detecting native schemas.



## SOLUTION

### Log Analytics at Scale

ChaosSearch empowers customers to Know Better™, activating the data lake for analytics. Unlike traditional log management, cloud management, and APM/observability tools, ChaosSearch indexes all log data in your cloud object storage, as-is, without the need for any data transformation or data movement.

ChaosSearch commonly replaces the Elasticsearch stack, yielding massive cost-performance improvements (up to 80% TCO savings), and making all your log data available without any data movement or behavior change needed from end users.

## CHAOSSEARCH BENEFITS FOR LOG ANALYTICS

### Management Simplicity

ChaosSearch is a fully managed service that eliminates the need to add, deploy, and manage hardware and software. ChaosSearch features a stateless architecture that delivers 99.999% uptime and no need to backup/restore ChaosSearch. The containerized nature of the architecture allows worker nodes to spin up and down and fail without affecting the user experience.

### Game Changing Price Performance

The unique ChaosSearch architecture and technologies consume far fewer resources than a comparable ELK stack - providing cost savings of up to 80%. And the patented indexing technology compresses data in your cloud object storage making it completely searchable at a dramatically lower cost.

### Performance and Timely Insights at Scale

The patented Chaos Cache is a non-memory caching system that returns queries in seconds at Petabyte scale. And Chaos Cache™ allows for infinite storage for queries and results. Compute pooling allows for defined groups of users or queries to have guaranteed compute resources while not affecting ad hoc queries. Queries won't get starved for execution time and users get faster time to insights.

### No Data Movement or Transformation

With ChaosSearch, there is no data movement, transformation, or schema management. The Chaos Refinery® cleans, prepares, and virtually transforms data directly in your Amazon S3. ChaosSearch also features a built-in Kibana interface and supports Elasticsearch APIs so you can use tools you know and love and quickly create more and better insights from your data.

**ChaosSearch's revolutionary approach supports the cloud environments' scalability and economic advantages so IT operations teams can break the endless cycle of break/fix, retain more log data at lower cost, and give users across the organization the data they need to generate impactful insights.**

*"With the team at ChaosSearch as our partner, we don't have to worry about the reliability of our log data or management. We are realizing extra engineering cycles internally to focus on product features that differentiate us—the way it should be."*

**Jason Standiford, VP**  
Engineering

**REVINATE**



## ABOUT CHAOSSEARCH

ChaosSearch empowers data-driven businesses like Blackboard, Equifax, and Klarna to Know Better™, delivering data insights at scale while fulfilling the true promise of data lake economics. The ChaosSearch Data Lake Platform indexes a customer's cloud data, rendering it fully searchable and enabling data analytics at scale with massive reductions of time, cost and complexity. The Boston-based company raised \$40M Series B in December 2020 and is hiring to support its hyper growth.

For more information, visit [ChaosSearch.io](https://ChaosSearch.io) or follow us on Twitter @ChaosSearch and LinkedIn.