Data Modernization

Global Financial Services

Finance

Our client, a Global Financial Services Leader, is driving an ambitious digital transformation program. The organization is moving toward a cell-based architecture; a modular, scalable framework designed to enhance resilience, efficiency, and innovation.

A core part of this transformation is the creation of a new Operational Data Store (ODS) capable of handling real-time event data while ensuring consistency, performance, and cost efficiency at scale. To achieve this, the client needed a modern event-processing backbone that could power materialized views and real-time insights across the business.

Utilized Technology Stack:

Database: MongoDB Atlas

Cloud Platform: AWS (EKS, MSK, Glue Schema Registry)

Backend Framework: Java Spring Boot

Event Streaming: Apache Kafka, Kafka Streams, MSK Connect

Infrastructure as Code: Terraform, Helm, GitHub Actions

Monitoring & Observability: OpenTelemetry, Prometheus, Loki, Tempo, Grafana

Testing: Python API and load testing suite

Review of the Challenges

Over years of growth and acquisitions, the client's systems had become complex and fragmented. Data was spread across multiple services, each with different levels of reliability and scalability. Maintaining operational stability while scaling globally had become increasingly difficult.

The goal of this Proof of Concept (PoC) was to explore how MongoDB Atlas could act as the Event Sink within the client's evolving cell-based architecture. The system needed to demonstrate how event-driven workloads could be ingested, processed, and queried in near real time, all whilst maintaining resilience, high throughput, and observability.

Access restrictions in the hosting environment and shared AWS infrastructure presented additional challenges. The team had to navigate permission constraints, manage CI/CD pipelines without a dedicated DevOps role, and ensure full observability in an unfamiliar cloud setup.

Our Solution

gravity9 engineered a fully working Event Sink architecture that combined MongoDB Atlas, Kafka, and AWS EKS into a seamless, event-driven system.

Our approach focused on building a lightweight but powerful platform composed of three core services:

Event Generator: A Spring Boot application that produced synthetic event data and published it to Kafka topics for ingestion and testing.

Kafka Streams Processor: A service that transformed and unified events, ensuring data consistency before pushing them to downstream topics.

Event API: A RESTful interface that allowed users to query, update, and manage event data stored in MongoDB.

We deployed these services using Infrastructure as Code, implementing Terraform, Helm Charts, and GitHub Actions to automate builds, deployments, and rollbacks. This setup reflected a true cell-based model with each cell acting as a self-contained, resilient unit capable of independent operation and recovery.

To ensure observability, the system was instrumented with the OpenTelemetry stack including Prometheus, Loki, Tempo, and Grafana which provided real-time metrics, tracing, and performance dashboards.

Finally, the architecture was rigorously tested to simulate production-level traffic. The solution achieved a sustained throughput of 1,500 events per

second, processing more than 600,000 documents with consistent latency and reliability.

Our Approach

We began by working closely with the client's architects to understand their current ODS landscape and how the Event Sink could fit into the larger cell-based framework. Our team then designed a modular deployment pattern that could scale across multiple cells while maintaining fault tolerance and high availability.

Using Terraform and Kubernetes, gravity9 built a fully automated CI/CD pipeline covering infrastructure provisioning, container builds, Helm deployments, and environment validation. Each service was developed with resilience in mind and built to recover from failures, restart gracefully, and maintain data integrity.

Throughout the engagement, gravity9 focused on performance, reliability, and transparency. Our engineers introduced a full observability layer from scratch, allowing the client to monitor every stage of event processing from Kafka ingestion through to MongoDB persistence, and all in real time.

Subsequent Outcomes

The Proof of Concept proved that MongoDB Atlas could efficiently serve as the Event Sink within a cell-based architecture, supporting event-driven operations at scale.

- The system achieved all key performance benchmarks:

- Sustained throughput of 1.5k events per second.

- Real-time event ingestion and transformation.

- Stable autoscaling and failure recovery within Kubernetes.

- Full-stack observability and traceability.

The success of this engagement has opened the door for the next phase; an MVP build focused on production deployment and extending the event-sink model across additional use cases within the client's digital ecosystem.

gravity9 demonstrated how a modern event-driven architecture can deliver resilient, real-time insights at scale for a complex financial ecosystem.

Visit our Insights page for more articles about emerging technology trends, the Finance Industry, interviews, and more!