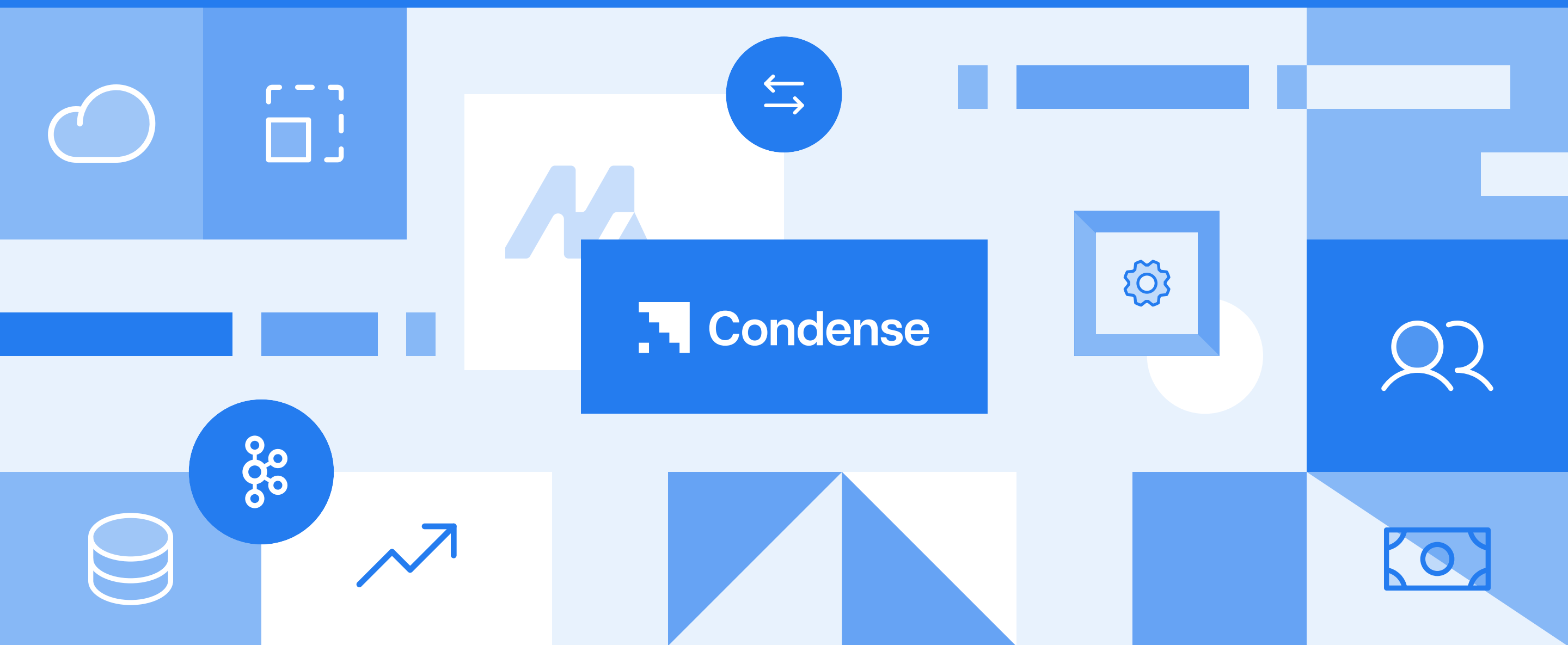


 Comparison

Data in Motion, Simplified at Scale

Evaluating Condense vs AutoMQ
for Enterprise Real-Time Data Streaming



 **Condense**

ZELIOT

Presented By :
Zeliot, for their platform [Condense](#): a Kafka- native, fully
managed streaming platform deployed in your cloud

 |  |  |  | 

All Rights Reserved 2026 @ Zeliot

How Does **Condense** Compare vs AutoMQ?

Condense and AutoMQ both address the complexities of cloud-native streaming, but they solve for entirely different outcomes. AutoMQ is an infrastructure-focused storage engine designed to slash Kafka costs by moving data to S3, but it remains a transport layer that introduces significant tail latencies and requires you to manage separate clusters like Flink or Spark to actually use your data. Condense solves the entire application lifecycle by integrating high-performance brokers with a native execution layer for your Java, Python, or Go code. It autonomously scales your brokers and your logic in lockstep, eliminating the need for separate processing infrastructure and the "latency tax" of moving data between decoupled storage and external compute

> Architecture

FEATURE	CONDENSE	AUTOMQ
System Design	A unified environment that bundles Kafka, Kubernetes, and the App Runtime into one integrated stack	Decoupled "Diskless" architecture that replaces local disks with S3/Object Storage for a stateless Kafka experience
Unified Platform	Unified Application Fabric: Merges the Kafka engine with a native event-driven microservice runtime. It is an "Execution Fabric"	Infrastructure-Only: A cloud-native storage engine; remains a transport layer requiring external tools for business logic
Scaling Scope	Full-Stack Autonomy: Automatically scales brokers, connectors, AND your custom-code transforms (Java/Python/Go) based on real-time consumer lag	Storage-Centric Scaling: Extremely elastic broker scaling (0 to GBps in minutes), but scaling the code that processes the data is a separate engineering task
Cloud Storage	Native Object Storage: Direct offloading to S3/GCS/Azure Blob; data stays in your buckets for infinite retention	S3-Native Primary Storage: Entirely replaces EBS with S3/Object storage to eliminate the "3x replication tax" of traditional Kafka
Enterprise BYOC	100% BYOC Native: Specifically engineered to run in your VPC to eliminate "SaaS" networking taxes and maintain 100% data sovereignty	Cloud-First BYOC: Designed for the customer Cloud via Marketplace/CFT; optimized to reduce cross-AZ data transfer costs
Performance	Serverless Scale and Zero-Hop: Running GBps+ enterprise data workload and autonomous scale for spikes and variations. Application logic runs "local" to the broker, eliminating the network latency of external processors	Elastic Performance: Optimized for high-throughput bursts and instant cluster resizing without partition rebalancing

> Ecosystem

FEATURE	CONDENSE	AUTOMQ
Operational Effort	Zero-Ops Application: Eliminates "Microservice Sprawl." You simply write the logic; the platform autonomously runs, scales, and monitors the entire application pipeline	Simplified Infrastructure: Removes Kafka disk management and rebalancing, but leaves you to manage the external apps/clusters that process the data
Industry Readiness	Dual-Layer Ecosystem: Access a vast library of generic connectors alongside industry specific domain-aware transforms for Mobility, IoT, and FinTech (e.g., VIN parsing, Geofencing) out of the box	Horizontal Generalist: Optimized for raw byte-moving at scale. Any industry-specific logic must be engineered from scratch by your team
Apache Kafka API	100% Native: Built on upstream Kafka 4.0/ KRaft	100% Native: Direct code-level compatibility with Kafka 3.x/4.0 and KRaft
Logic Execution	Managed App Runtime: Custom transforms run in a managed, serverless-style environment. Condense handles containerization, state management, and orchestration	External Execution: Data must be piped to external clusters (Flink, Spark, or K8s microservices) that your team must manage separately
Dev Experience	Integrated AI IDE: Build, test, and deploy production Java/Python logic natively. Includes Agentic AI for root-cause analysis and code generation	Standard Tooling: Works with existing Kafka UIs and CLI tools; lacks an integrated logic development environment
Observability	Inbuilt Full-Stack: Trace an event from ingestion through your custom code to the final sink in a single, unified dashboard	Storage Observability: Superior metrics for S3 throughput and storage latency, but lacks integrated application-layer tracing
Flink Integration	Native Runtime Alternative: Replaces the need for Flink for most use cases	External: Requires Flink or similar external engines for any stateful stream processing

> Support & Compliance

FEATURE	CONDENSE	AUTOMQ
Access Experts 24/7	Direct Engineering: Built in support widget to raise ticket. Condense offers support priority based TAT for resolution	Enterprise Support: Mature support model focused on mission-critical broker operations
Cloud Availability	AWS, GCP, Azure (Customer Subscription)	AWS, GCP, Azure (Customer Subscription)
Security	Own Cloud -Sovereign: Inherits your IAM and security groups; 100% data localization	Customer Cloud: Strong security posture with air-gapped deployment options
SLAs	99.95%: Covers the brokers, the processing logic, and the connectors under one SLA	99.9% - 99.99%: Varies by deployment; focused on broker availability and storage durability
Pricing / TCO	Compute-Based: Simple, predictable vCPU/hr billing. Zero "Connector " or per-partition fees	Usage-Based Infrastructure: Pricing focused on data throughput and S3 costs; significantly reduces cloud bills by removing EBS

Why Switch to Condense?



TRANSITION FROM MESSAGE BROKER TO APPLICATION ENVIRONMENT

AutoMQ re-engineers the broker for the cloud by moving storage to S3, but it remains a transport and storage layer. To actually use the data, teams still face the overhead of managing external processing clusters, such as Flink, Spark, or Kubernetes microservices

💡 Condense merges these layers. It provides a managed environment that hosts both the data and the business logic as a single entity. This allows developers to ship production-grade code directly on the stream without the burden of building and maintaining a separate processing tier



FULL-STACK AUTONOMOUS SCALING

In an AutoMQ architecture, scaling is highly elastic at the broker level, but scaling is restricted to that infrastructure. If the processing logic (external apps or consumers) starts to lag during a traffic spike, manual intervention is required to adjust compute resources for those external services

💡 Condense introduces Autonomous Scaling for the entire pipeline. It monitors real-time consumer lag and throughput at the event level. When data volume surges, the platform automatically provisions compute for custom Java, Python, or Go transforms. It ensures processing power stays in sync with data volume and automatically scales back down to optimize costs once the surge passes

Why Switch to Condense?



PRIVATE CLOUD SOVEREIGNTY (BYOC DEPLOYMENT)

While AutoMQ allows for data to be stored in your own S3 buckets, but it primarily focuses on the storage layer. While the data stays in your account, you must still manage the security and networking for the external compute clusters where your application logic actually runs



Condense is built from the ground up to be a 100% BYOC Native complete platform. It deploys directly into the customer Cloud (AWS, Azure, or GCP), keeping 100% of the data and the application execution within the customer's control. This allows for the inheritance of existing security policies and encryption keys while utilizing existing cloud enterprise credits for both the messaging and the application compute



VERTICAL SOLUTIONS VS GENERAL-PURPOSE PIPES

AutoMQ is a horizontal tool designed for infrastructure efficiency and raw byte-moving, meaning all industry-specific logic must be engineered from scratch



Condense provides a Verticalized Ecosystem. It includes pre-built, domain-aware transforms for industries like Mobility (e.g., VIN parsing, trip decoding) and IoT. By using these pre-tuned assets, teams can bypass months of custom development and move from prototype to production significantly faster



UNIFIED DEVELOPMENT ENVIRONMENT AND LATENCY CONTROL

AutoMQ's S3-native approach can introduce significant tail latencies (100ms-500ms) because data must travel to object storage and then out to an external processing cluster







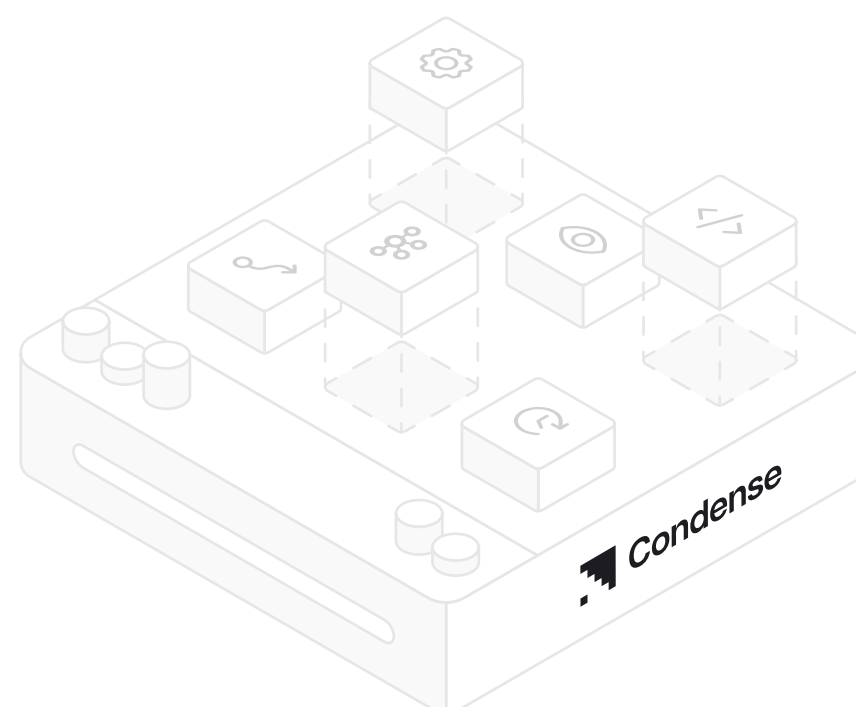
Condense embeds a Full-Code IDE directly into the platform and utilizes a Zero-Hop architecture. Developers can write, test, and deploy production transforms in standard languages where the logic runs "local" to the broker. This eliminates the network latency of external processors and is supported by an AI-assisted layer that handles root-cause analysis, reducing the manual effort required to keep complex pipelines healthy



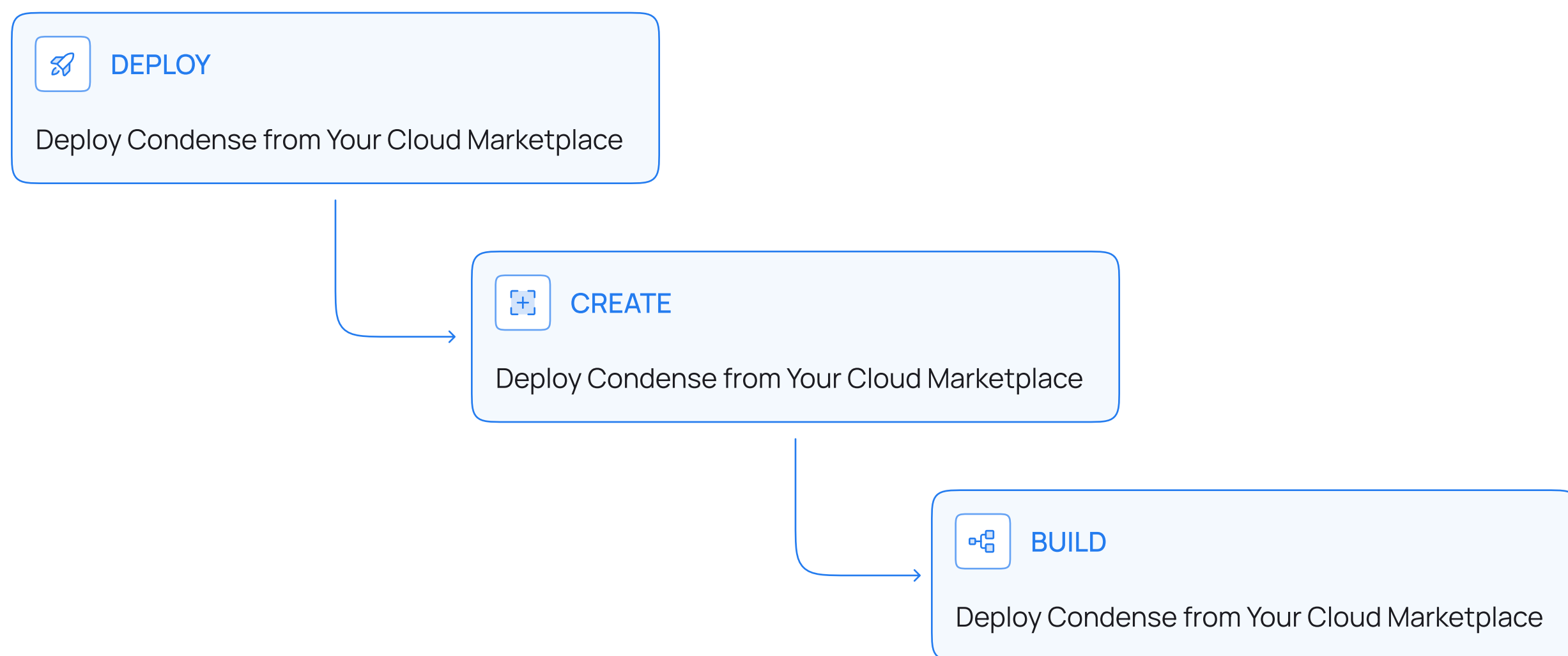
Vertical Data Platform for Real-Time Businesses

Condense understands vertical data by design and connects directly to real time data sources like vehicles or GDS platforms. Its purpose-built industry specific connectors and transformations accelerate the development of real-time vertical use case

-  Accelerate the path from idea to production
-  Rapidly realize and deploy industry use cases
-  Enterprise scale with native governance
-  TCO ↓ through vertical industry-first approach



Get Started with Condense in 3 Simple Steps



Most platforms focus on streaming data, Condense focuses on understanding it. It's not just managed Kafka; it's Kafka with context. Infused with domain intelligence, **Condense transforms raw streams into decisions, automations, and impact**, because the right Kafka platform should understand what it's moving

Hope this analysis sparked a new vision for your data architecture!

Stick Around, It Gets Better



[Read Our Blogs](#) →



[Read Our Documentation](#) →



[Read Our eBooks](#) →

We're Ready When
You Are. Let's Talk.

[Schedule a Meeting](#)

Don't Miss a
Thing - Come Along

