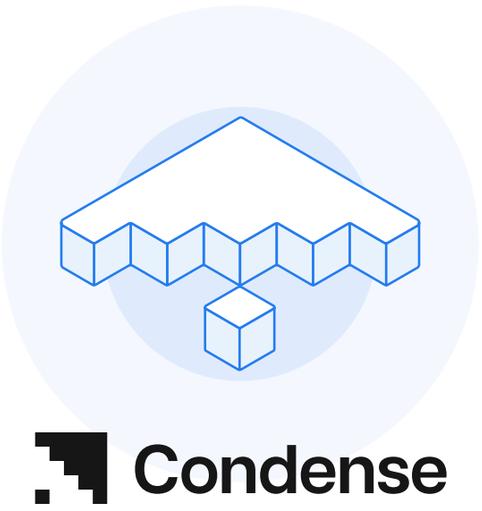
Why Condense Is a Better Choice Over OSS Kafka



TRULY CLOUD-NATIVE DESIGN

- ⊗ Elastic scaling is manual; requires capacity planning and risk of over provisioning.
- ⊗ No native tiered storage; long-term retention needs complex custom solutions
- ⊗ High availability must be architected manually; no SLA
- ⊗ Multi-AZ deployment possible but manual; complex
- ⊗ ZooKeeper partly removed only in new versions; user must handle metadata ops
- ⊗ **No automatic patching**; user must handle upgrades manually

- ✓ Built-in auto-scaling optimizes resource usage seamlessly
- ✓ Native tiered storage reduces storage costs while retaining long-term data
- ✓ 99.95% uptime SLA with built-in failover and health monitoring
- ✓ Multi-AZ resiliency configured out-of-the-box
- ✓ Metadata management fully abstracted; no ZooKeeper burden
- ✓ No-touch, zero-downtime patching and upgrades managed by Condense

DEVELOPER PRODUCTIVITY

- ✓ Distributed event streaming engine (Apache Kafka core).
- ⊗ Kafka Connect available; requires manual connector integration.
- ⊗ Kafka Streams client-side library; user must manage compute.
- ⊗ No inbuilt IDE; developers work outside Kafka (custom apps).
- ⊗ No AI coding assistance; manual coding, error handling.

- ✓ Comes with fully managed distributed event streaming engine at core (Kafka)
- ✓ Pre-built production-ready connectors natively available.
- ✓ Stream apps supported via IDE + platform-deployment
- ✓ Inbuilt IDE to write, test, and deploy logic with versioning.
- ✓ AI-assisted code development: assists, error detection, auto-completion inside the IDE

Why Condense Is a Better Choice Over OSS Kafka

kafka	Condense
SECURITY & COMPLIANCE	
⊗ ACLs available but needs manual configuration	✓ ACLs can be configured easily inside the platform
⊗ No native RBAC; external LDAP/SSO integrations needed	✓ Native RBAC with fine-grained access at resource level
⊗ No built-in audit logging; must integrate third-party solutions	✓ Structured, searchable audit logs automatically maintained
⊗ TLS setup required manually; secret protection depends on external config	✓ End-to-end encryption built-in; secrets are protected by design
⊗ Private networking must be built by user (VPC, VPN peering)	✓ Native private networking via VPC/VNet peering
⊗ Compliance depends on user infra (not OSS Kafka itself)	✓ Condense platform is ISO 27001 certified by default
EFFICIENT OPERATIONS	
⊗ No GUI for management; heavily CLI/API driven	✓ Full GUI-driven cluster management via Condense UI
⊗ JMX-based metrics; complex and manual to extract useful insights	✓ Metrics API simplifies access to topic, consumer group, cluster KPIs
⊗ External monitoring setup needed (Prometheus, Grafana dashboards)	✓ Built-in integrations with Datadog, Grafana, Prometheus for observability
⊗ No health monitoring or auto-alerting built-in	✓ Health+ service for proactive monitoring and alerting
DEVOPS AUTOMATION	
⊗ Limited Admin APIs; requires direct Kafka API knowledge	✓ Full Admin REST APIs for cluster and topic management
⊗ K8s deployment possible but needs Helm chart customization; no declarative control	✓ Condense Operator provides declarative, native Kubernetes control
⊗ No marketplace deployments; manual IaaS setup	✓ Available on AWS, Azure, GCP marketplaces
⊗ Manual intervention needed for partition rebalancing	✓ Self-balancing clusters automatically rebalance partitions for performance

Why Condense Is a Better Choice Over OSS Kafka

kafka	Condense
STREAM PROCESSING & INTEGRATION	
⊗ No visual pipeline builder; pipelines must be built using custom code	✓ Stream Designer: Drag-and-drop visual pipeline builder
⊗ No no-code/low-code modules; everything must be coded	✓ Prebuilt no-code utilities like conditional logic, windowing, splits, joins
⊗ No native SQL query engine for Kafka OSS; KSQL requires separate setup	✓ KSQL query engine natively available to explore live streams
⊗ No built-in way to download queried data	✓ Ability to download query results directly from the UI
DATA GOVERNANCE	
⊗ Schema Registry available via separate installation	✓ Schema Registry built-in and fully managed
⊗ No broker-side schema validation; only at producer level	✓ Broker-side schema validation enforced at topic-level granularity
⊗ No data pipeline approach or end-to-end visibility	✓ Data Pipeline builder: Visual, end-to-end data movement design and tracking
⊗ Data contracts not enforced broker-side.	✓ Ensures strong governance and compatibility
GLOBAL RESILIENCE	
⊗ No guaranteed SLA	✓ 99.95% SLA covering software + infrastructure
⊗ Multi-AZ or Multi-Region must be manually built using MirrorMaker 2	✓ Multi-AZ and Multi-Region deployments natively supported
⊗ MirrorMaker 2 replication is manual and operationally heavy.	✓ MirrorMaker 2 managed as part of platform services
EXPERT SUPPORT	
⊗ Community-based support via forums, Slack, GitHub	✓ 24x7x365 enterprise-grade support included
⊗ Professional services only available from third-party consultancies	✓ Professional Services available for technical consulting