

USE CASE 05 · FOR SRE · DEVOPS · PLATFORM

LOGS AT LINE RATE. QUERIES IN MILLISECONDS.

Operational intelligence is where AI transformation meets pager duty. Agents that remediate, enrich, and triage need retrieval at agent cadence — not at log-pipeline cadence. XERJ ingests at line rate and returns aggregations in milliseconds on the same data.

80K DOCS/S SUSTAINED · NO WRITE-STALLS · 89× FASTER TERMS AGGS

02 · AI MEETS ON-CALL

AGENTS NEED EVIDENCE. AT AGENT SPEED.

Your AI roadmap names on-call copilots, remediation agents, and runbook automation. Every one of them fails on the same bottleneck: the logs they need to read are in a store that was never designed to answer an agent's question in milliseconds. The SRE team pays the price twice — once to the incumbent log vendor, once to the embedding provider for context retrieval.

INGEST VS QUERY	Log pipelines optimize for throughput. Query engines optimize for latency. Most systems pick one. Agents need both — high-volume ingest AND sub-second aggregations on the same data.
ILM PUZZLES	Index Lifecycle Management (ES), retention buckets (Loki), storage tiers (Splunk) — every vendor has their own version of "where does old data go?" All require tuning; all misbehave under ingest spikes.
SHARD TUNING	Shard count, shard size, heap ratios, index rollover policies — the SRE team's runbook is half operational ceremony. Junior engineers spend quarters learning the vendor-specific dark arts.
AGENT BLIND SPOTS	An agent asking "what services saw 429s in the last 5 minutes" needs a terms agg. At 30 ms p95 a human accepts it; at 30 ms per call an agent with a 10-hop investigation accepts nothing.

03 · INGEST + QUERY ON ONE PLANE

80 K/S IN. MS AGGS OUT. SAME NODE.

XERJ is the first data plane engineered so that ingest throughput and query latency are not a tradeoff. Columnar log storage, lock-free readers, single-process ingest — the same node that absorbs 80 K docs/sec from Vector also serves a 0.4 ms p95 terms aggregation to an agent.

INGEST

80 K
DOCS/SEC
ONE CLIENT

Sustained · diurnal · single-producer. Scales on sharded ingest (roadmap Q3).

QUERY

89×
FASTER TERMS AGG
VS ES 8.13

0.4 ms p95 vs 35.7 ms. Agents iterate through evidence without budget-burn.

STORAGE

2.8×
LESS DISK
PER M EVENTS

85 MB vs 240 MB. Longer hot retention on the same hardware. Agents see more history.

STALLS

0
WRITE-STALLS
UNDER SIZING

Lock-free readers mean flush never blocks agent queries. Agent context stays fresh during ingest spikes.

04 · OPS-INTEL BLUEPRINT

ONE PIPELINE. ONE API.

A three-node XERJ cluster acting as the operational-intelligence layer for an engineering org. Ingest from the fleet; query by humans and agents over the same surface.

FLEET

Application logs · system logs · k8s events · cloud audit · network



▼ one Otel / Vector agent per host

Unified ingest (OTLP · ES _bulk · JSON / NDJSON · syslog)



XERJ DATA PLANE 3-node HA · 80 K events/sec sustained

COLUMNAR LOGS (delta-ts · dict · varint · ZSTD)

FTS index for free-text search on message bodies

VECTOR index for log-embedding similarity (anomaly lookup)

pre-computed term histograms (roadmap Q2) – fast agg path



▼ ES 8.13 wire · explain-plan · /metrics

CONSUMERS

SREs Grafana dashboards · Kibana log search · CLI

Agents runbook agent · RCA agent · on-call copilot
· hybrid retrieval: keyword filter + vector similarity

Alerting Alertmanager-compatible · anomaly-detection hooks (roadmap)

RETENTION

Hot (SSD) 14 d · interactive + agent queries

Warm (HDD) 30 d · aggregation-only queries

Cold (S3) 90 d · incremental encrypted snapshots

SIZING · 3 NODES · 80K SUSTAINED · ~2.5 TB/NODE

05 · WHY INGEST + QUERY COEXIST

ENGINEERING CHOICES THAT MAKE IT WORK.

Six choices that let one node absorb 80 K events/sec while serving millisecond aggregations to agents on the same data.

LOCK-FREE READERS	ArcSwap segment view means the writer never blocks a reader. A 24 h ingest surge does not stall the agent's query. No GC pauses.
SINGLE-PROCESS INGEST	Parser + writer in one process. One fsync, one network hop, end-to-end back-pressure. No Logstash to tune · no Kafka buffer to operate · no cross-service retry logic.
COLUMNAR BY DEFAULT	85 MB per 1 M SIEM events post-ingest · 2.8× smaller than ES on the same corpus. Delta-of-delta timestamps · dictionary keywords · varint integers · ZSTD blocks.
TERM HISTOGRAMS	Pre-computed per-column histograms at ingest time (roadmap Q2). Terms aggregations read the answer instead of scanning the column. The 89× NGINX number depends on this path.
MMAP SHARED CACHE	One OS page cache backs every reader. No repeated mmap() syscalls. Hot data stays hot across restart, across HPA scale-up, across rolling deploy.
EXPLAIN-PLAN FOR AGENTS	Agents call /v1/indices/{idx}/explain-plan before the real query to budget cost. Runaway agents fail loud, not expensive. Per-tenant quotas enforce the budget.

06 · MEASURED VS ES 8.13

INGEST + QUERY. SAME NODE.

Eight operational queries representative of AI-assisted on-call. p95 on 1 M events. XERJ single node vs. 4-node ES baseline.

QUERY / METRIC	XERJ	ES 8.13 (4-NODE)
NGINX TERMS AGG · 5 DISTINCT	0.4 ms	35.7 ms — 89×
TOP ERROR CODES · 1 M EVENTS	0.4 ms	29.8 ms — 74×
KEYWORD SEARCH · METHOD=GET	0.3 ms	2.0 ms — 6.7×
SUSTAINED INGEST · ONE CLIENT	80 K docs/s	typically 20–40 K
DISK · 1 M EVENTS POST-INGEST	85 MB	240 MB — 2.8×
COLD START	50 ms	~15 s JVM warm-up
IDLE MEMORY	400 MB	8.5 GB — 21×
WRITE-STALLS UNDER DIURNAL PEAK	0	occasional · depends on shard tuning
AGENT IMPACT	An on-call agent doing a 10-query RCA completes in under 10 ms of retrieval budget. The same chain over ES 8.13 on the same corpus routinely burns 300 + ms in aggregations alone.	
SOURCE	SIEM_BATTLE_2026-04-14_184900.UTC.md · BATTLE_20M_2026-04-15.md · BATTLE_SLA_RESULTS_2026-04-15_21-32-15.md	

07 · YOUR OPS STACK

VECTOR.
OTEL.
AS-IS.

The collectors, dashboards, and alerting tools your SRE team already operates work with XERJ out of the box — via ES wire, OTLP, or Prometheus.

CATEGORY	TOOLS	STATUS
COLLECTORS	OpenTelemetry Collector · Vector · Fluent Bit · Fluentd · Filebeat · Metricbeat	GA
DASHBOARDS	Grafana ES datasource · Kibana · custom UI via ES query API · real-time log-tailer	GA
ALERTING	Alertmanager-compatible · Grafana alerting · PagerDuty / Opsgenie webhooks · anomaly detection (roadmap Q4)	GA
RUNBOOK AGENTS	LangChain · LlamaIndex · Anthropic Agent SDK · OpenAI Agents — ES retriever, hybrid retrieval, explain-plan	GA
INCIDENT MGMT	PagerDuty · Opsgenie · Incident.io · ServiceNow via webhook — auto-enrichment from XERJ context	GA
K8S	Helm chart · Kubernetes Operator (roadmap Q2) · StatefulSet per data-plane role · HPA on CPU + ingest rate	ROAD Q2
IAC	Terraform module (roadmap Q2) · Pulumi · Ansible · Helm chart	ROAD Q2
CHAT-OPS	Slack bot · Microsoft Teams · generic webhook · /summon agent that retrieves live context on request	GA
TOPOLOGY	ServiceNow CMDB · AWS/GCP/Azure inventory · OpenTelemetry resource attributes · auto-enriched on ingest	GA

08 · DAY TWO

ROLLING DEPLOYS. BY DEFAULT.

Day-two operations is where AI-ops platforms usually lose credibility. XERJ is engineered for rolling restart, hot-reload config, and segment-level backup from the first binary.

HEALTH · READINESS	/healthz for liveness. /readyz holds until WAL replay completes — rolling restart fleet-safe. HPA autoscales on CPU + ingest queue depth.
METRICS OUT	Prometheus on /metrics. 13 counters / histograms / gauges across ingest, query, flush, cache. Per-instance registry · multi-tenant-safe.
CONFIG	Single TOML file · XERJ_CONFIG env · hot-reload via SIGHUP on non-structural changes. Schema validated on load; broken config refuses to start, never to serve.
UPGRADES	Rolling restart supported. Segment + WAL formats forward-compatible across minor versions. One-minor-version rollback window. Helm rolling update in-place.
BACKUP · RESTORE	Snapshots to S3 / GCS / Azure Blob. Incremental · encrypted · restorable at segment granularity. xerj snapshot create list restore.
CAPACITY	Sizing guide: ~80 K events/sec sustained per node. Scale horizontally from 3 to 12 nodes. Sharded ingest (roadmap Q3) extends to hundreds of K events/sec per cluster.

09 · FIVE PHASES · ONE YEAR

FROM PILOT TO AGENTS.

Start with the highest-volume log stream. End with on-call agents operating on unified retrieval across logs, traces, and runbook embeddings.

01	PILOT WEEKS 1–2	Single node · highest-volume log stream routed via Vector · reproduce BATTLE_20M on your hardware · first Grafana dashboard points at XERJ.
02	SHADOW WEEKS 3–6	3-node HA · dual-write from Vector · Kibana / Grafana on both · compare ingest throughput, query latency, disk footprint on production traffic.
03	LOGS PRIMARY QUARTER 2	Primary = XERJ for log workloads. Decommission shadowed log store. First bill line removed. SRE runbooks point at XERJ; on-call runs on XERJ.
04	ADD AGENTS QUARTER 3	Deploy runbook + RCA agents over hybrid retrieval. Embed runbooks + postmortems as vectors in the same platform. MTTR drops as agents do tier-1 triage.
05	UNIFY QUARTER 4	Metrics and traces migrate onto XERJ (see Unified Observability brief). One platform, one query surface, one on-call rotation for the data layer.

REVERSIBLE · ES + OTLP MEAN NO LOCK-IN AT ANY PHASE

10 · NEXT STEPS

RUN IT. ON YOUR FLEET.

A two-week pilot on one high-volume log stream. We send the binary, Vector dual-write config, and the BATTLE_20M reproduction pack. You bring the stream and a latency baseline.

SRE / PLATFORM CONTACT

HELLO@XERJ.AI

XERJ.AI · XERJ.COM · XERJ.DEV

WHAT YOU GET	Private binary · Vector dual-write config · Grafana dashboard JSON · BATTLE_20M reproduction scripts · RCA-agent starter.
WHAT YOU BRING	One high-volume log stream · its ingest rate baseline · one terms agg or keyword query with its p95.
COMPANION BRIEFS	xerj-usecase-observability.pdf for unified obs · xerj-usecase-security.pdf for SIEM · xerj-tech-brief.pdf for architect detail